Bayesian Spatial Field Reconstruction with Unknown Distortions in Sensor Networks

Bayesian Spatial Field Reconstruction with Unknown Distortions in Sensor Networks

Qikun Xiang[1], Ido Nevat [2] and Gareth W. Peters[3],

[1] School of Physical and Mathematical Sciences, Nanyang Technological University, Singapore

[2] TUMCREATE, Singapore

[3] Department of Actuarial Mathematics and Statistics, Heriot-Watt University, Edinburgh, UK

Abstract

Spatial regression of random fields based on potentially biased sensing information is proposed in this paper. One major concern in such applications is that since it is not known *a-priori* what the accuracy of the collected data from each sensor is, the performance can be negatively affected if the collected information is not fused appropriately. For example, the data collector may measure the phenomenon inappropriately, or alternatively, the sensors could be out of calibration, thus introducing random gain and bias to the measurement process. Such readings would be systematically distorted, leading to incorrect estimation of the spatial field. To combat this detrimental effect, we develop a robust version of the spatial field model based on a mixture of *Gaussian process* experts. We then develop two different approaches for Bayesian spatial field reconstruction: the first algorithm is the Spatial Best Linear Unbiased Estimator (S-BLUE), in which one considers the quadratic loss function and restricts the estimator to the linear family of transformations; the second algorithm is based on empirical Bayes, which utilises a two-stage estimation procedure to produce accurate predictive inference in the presence of "misbehaving" sensors. In addition, we develop the distributed version of these two approaches to drastically improve the computational efficiency in large-scale settings. We present extensive simulation results using both synthetic datasets and semi-synthetic datasets with real temperature measurements and simulated distortions to draw useful conclusions regarding the performance of each of the algorithms.

Keywords: Sensor Networks, Gaussian Process, Spatial Linear Unbiased Estimator (S-BLUE), Empirical Bayes, Cross Entropy method (CEM), Iterated Conditional Modes (ICM)

I. Introduction

In recent years, Wireless Sensor Networks (WSNs) have attracted considerable attention due to their applications in environment monitoring [1], [2], forecasting [3], surveillance [4], event detection [5] and tracking [6]. For example, the United States Environmental Protection Agency (EPA) proposed to promote the use of sensor networks for air quality monitoring [7]. In this paper, we focus on environmental monitoring applications in which a WSN consists of a collection of spatially distributed sensor nodes with limited energy and communication bandwidth. The sensors make observations of spatial physical phenomena (e.g. the concentration of air pollutants such as carbon monoxide and ozone, temperature, humidity, etc. [8], [2]) and communicate the observations to a Fusion Center (FC) [9]. The FC then reconstructs the spatial phenomena from these observations at any spatial location of interest, based on which decisions can be made and actions can be performed.

In many cases, the sensor nodes used to collect the spatial information are unreliable and distort the spatial information. These distortions should be accounted for in spatial field reconstruction. It is therefore crucial to assess and guarantee the veracity, quality and reliability of collected data [10], [11]. There are multiple reasons for such a behaviour and here we list three common reasons:

1) Uncalibrated sensors: uncalibrated sensors, if ignored, can lead to severe degradation of the quality of the fields estimation [12]. Traditionally, sensors are calibrated in a controlled environment where the physical input is known and then their performance in a given calibration range is tested and verified over certain operating ranges of the environment, before such WSNs are deployed. This is infeasible for large-scale WSNs due to the prohibitive cost as well as inhomogeneity in deployment schedules. Thus, the calibration has to be done through the so-called *blind* or *self-calibration* techniques [13]. In addition, the reliability of sensor can deteriorate over time [8], making it very challenging to guarantee the quality of information even for an *a priori* calibrated network. Among the classical calibration models, the *gain-offset* response model is widely-used [14], [15]. We consider the problem of jointly estimating the calibration parameters of individual sensors as well as the spatial field values [16], [17].
2) Compromised sensors due to malicious intent: the sensors may be physically compromised such that the sensor observations are maliciously altered to disrupt the operation of the WSN [18]. One common type of attacks is the *Byzantine attack* in which a hostile attacker

compromises a part of the sensor network in such a manner that the Fusion Center has imperfect knowledge about whether a sensor node has been compromised [19], [20]. In such a case, there may be erroneous information incorporated into the observations from compromised sensors.

3) Unintentional misuse of sensors: in many cases data collection is done via crowd-sourcing in which private individuals install sensing stations in order to collect and share their data [21]. Since this type of data collection is not performed by professionals, in many cases, the sensors are not placed or used properly, thus introducing distortion into the measurements. Applications of crowd-sensing are becoming common recently due to the ubiquity of the Internet of Things (IoT) [22]. Those sensors could be stationary [2], [23], [24], [1], [25] or mobile [26], [27], depending on the application.

Many recent works such as [2], [28], [23], [24], [1], [25], [26] utilise the spatial-temporal correlation to reconstruct spatial physical phenomena at all locations. For example, [2] studies the placement of multi-type sensors in Gaussian spatial field to achieve optimal spatial field monitoring. To circumvent the threat to data reliability in environment monitoring systems, an estimation procedure was proposed in [28] to detect and exclude malicious sensing agents, while accurately performing spatial field reconstruction.

The main goal of this paper is to develop statistical procedures that reconstruct spatial fields using observations from sensors with possibly unknown distortions. We refer to such observations as *distorted observations*. We use Gaussian processes as the probabilistic model for spatial phenomena, and the sensors are assumed to follow the *gain-offset* distortion model with multi-modal priors to capture distortion characteristics resulted from different processes, such as natural deterioration, mis-calibration, malicious tampering, and unintentional misplacement or misuse.

The main contributions are as follows:

1) We develop a two-stage Bayesian inference algorithm that jointly infers the distortions of sensors and reconstructs the spatial field at all locations of interest. The algorithm estimates the distortion parameters in an empirical Bayes manner.
2) We derive the posterior distribution and the posterior predictive distribution of the model, and show that the exact computation of Bayes estimators is intractable.
3) We develop the Spatial Best Linear Unbiased Estimator (S-BLUE) for the model, which is highly computationally efficient.

4) We solve the optimization problem resulted from empirical Bayes estimation via two efficient methods, the Cross-Entropy method (CEM) and the Iterated Conditional Mode (ICM) method.
5) We analyse the computational time complexity of the proposed approaches and develop simple distributed versions of these approaches that are computationally more efficient and suitable for large-scale applications.
6) We perform synthetic data experiments as well as an experiment with real-world scenarios to validate our model and estimation procedures. The study with real-world scenarios uses a real temperature dataset from US EPA with synthetically generated distortions to show the real-world applicability of the model.

The remainder of the paper is organized as follows. We present our Bayesian sensor network model in Section II, which includes the prior distribution of the distortion parameters. In Section III, we derive the posterior distribution of the parameters as well as the posterior predictive distribution. Section IV introduces the S-BLUE and its properties. Section V introduces the approximation of the Bayes estimators via empirical Bayes, and shows that the maximization of the posterior distribution can be done through CEM and ICM. Section VI introduces the distributed approaches. In Section VII and Section VIII, we perform experiments using synthetic and real datasets. Finally, Section IX concludes the paper.

II. Sensor Network Model and Assumptions

We begin by presenting the statistical model for the spatial physical phenomena, followed by the system model. The following notational convention is used throughout this paper. Boldface upper case symbols denote matrices, boldface lower case symbols denote column vectors, and standard lower case symbols denote scalars or scalar-valued functions, unless otherwise specified. All vectors are column vectors unless otherwise stated.

A. Spatial Gaussian Random Fields Background

We model the physical phenomenon as spatially dependent continuous process with a spatial correlation structure. Such models have recently become popular due to their mathematical tractability and accuracy [1], [25], [29], [28], [3], [2]. The degree of the spatial correlation in the process increases with the decrease of the separation between two observing locations and can be accurately modeled as a Gaussian random field[1]. A Gaussian process (GP) defines a distribution

[1]We use Gaussian Process and Gaussian random field interchangeably.

over a space of functions and it is completely specified by the equivalent of sufficient statistics for such a process, and is formally defined as follows.

Definition 1. *(Gaussian process [30]): Let $\mathcal{X} \subset \mathbb{R}^d$ be some bounded domain of a d-dimensional real-valued vector space. Denote by $f(\mathbf{x}) : \mathcal{X} \mapsto \mathbb{R}$ a stochastic process parametrized by $\mathbf{x} \in \mathcal{X}$. Then, the random function $f(\mathbf{x})$ is a Gaussian process if all its finite dimensional distributions are Gaussian, where for any $m \in \mathbb{N}$, the random vectors $(f(\mathbf{x}_1), \ldots, f(\mathbf{x}_m))$ has multivariate normal distribution.*

We can therefore interpret a GP as formally defined by the following class of random functions:

$$\begin{aligned}
&\mathcal{F} := \{f(\cdot) : \mathcal{X} \mapsto \mathbb{R} \text{ s.t. } f(\cdot) \sim \mathcal{GP}(\mu(\cdot), \mathcal{C}(\cdot,\cdot)), \\
&\text{with } \mu(\mathbf{x}) := \mathbb{E}[f(\mathbf{x})] : \mathcal{X} \mapsto \mathbb{R}, \\
&\mathcal{C}(\mathbf{x}, \mathbf{x}') := \mathbb{E}\left[(f(\mathbf{x}) - \mu(\mathbf{x}))\left(f(\mathbf{x}') - \mu(\mathbf{x}')\right)\right] : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}\},
\end{aligned}$$

where at each point the mean of the function is $\mu(\cdot)$, and the spatial dependence between any two points is given by the covariance function (Mercer kernel) $\mathcal{C}(\cdot,\cdot)$ (see detailed discussion in [30]).

B. Sensor Network System Model

We begin by presenting the system model followed by the prior distribution specifications.

A1. Consider a random real-valued spatial phenomenon $f : \mathcal{X} \mapsto \mathbb{R}$ defined on the d-dimensional domain $\mathcal{X} \subset \mathbb{R}^d$.

A2. Consider a sensor network with N sensors that sense and transmit data to a Fusion Center (FC) over perfect communication channels. The spatial locations of the sensors, denoted $(\mathbf{x}_n)_{n=1:N}$ ($\mathbf{x}_n \in \mathcal{X}, n = 1, \ldots, N$), are known at the FC.

A3. The sensor n transmits $M_n \in \mathbb{N}$ observations to the FC. The observations $(y_{n,m})_{m=1:M_n}$ are generated according to the following *acquisition + distortion* mechanism:

$$\begin{aligned}
\widetilde{y}_{n,m} &= f(\mathbf{x}_n) + \epsilon_{n,m} \quad \text{(acquisition)} \\
y_{n,m} &= \mathcal{T}(\widetilde{y}_{n,m}; \boldsymbol{\psi}_n) \quad \text{(distortion)}
\end{aligned} \tag{1}$$

for $m = 1, \ldots, M_n$, where $f(\mathbf{x}_n)$ is the realisation of the random field at location $\mathbf{x}_n$, $\epsilon_{n,m}$ represents the additive random noise at the n-th sensor, and $\mathcal{T} : \mathbb{R} \mapsto \mathbb{R}$ is the distortion transformation function, parametrized by $\boldsymbol{\psi}_n$.

A4. The distortion transformation $\mathcal{T}$ has the following generic *gain-offset* form:

$$\mathcal{T}\left(u\,;\boldsymbol{\psi}_n=(a_n,b_n)^T\right):=a_nu+b_n, \tag{2}$$

where $a_n\in\mathbb{R}_+$ and $b_n\in\mathbb{R}$ represent the *gain* and *offset* of the n-th sensor, respectively. This *gain-offset* model has been widely used to describe sensor characteristics [31], [14], [15].

A5. We assume there are $K+1$ "categories" of possible distortion transformations. The auxiliary indicator random variable Z_n indicates the category to which each sensor's parameters $\boldsymbol{\psi}_n$ belong. We denote by $Z_n=0$, the *default distortion transformation* category, $\boldsymbol{\psi}_n=\boldsymbol{\psi}^0:=(1,0)^T$ i.e. no distortion ($\mathcal{T}(u,\boldsymbol{\psi}^0)=u$), whereas $Z_n=k, k\in\{1,\ldots,K\}$ indicates that the sensor n belongs to the k-th *non-default distortion transformation*.

C. Prior Distribution Specifications

P1. The spatial random field, f, is modelled as a Gaussian process (GP) F with a known mean function $\mu:\mathcal{X}\mapsto\mathbb{R}$ and a known covariance function $\mathcal{C}:\mathcal{X}\times\mathcal{X}\mapsto\mathbb{R}$, that is,

$$F\sim\mathcal{GP}(\mu(\,\cdot\,),\mathcal{C}(\,\cdot\,,\,\cdot\,)). \tag{3}$$

P2. The additive random noise, $\epsilon_{n,m}$, follows a normal distribution with mean zero and a fixed known variance ς^2,

$$\epsilon_{n,m}\overset{\text{i.i.d.}}{\sim}\mathcal{N}(0,\varsigma^2). \tag{4}$$

P3. We place a prior distribution on Z_n, denoted $\pi(Z_n)$, which is a categorical distribution given by

$$\pi(Z_n=k)=q_k^{(n)}\,,k=0,\ldots,K, \tag{5}$$

where $q_k^{(n)}\geq 0$ for $k=0,\ldots,K$, and $\sum_{k=0}^{K}q_k^{(n)}=1$.

P4. Each category of distortion characteristics has a distinct sub-population distribution, denoted π_k, which translates to the following. For $k=0,\ldots,K$,

$$(\boldsymbol{\psi}_n|Z_n=k)\sim\pi_k, \tag{6}$$

where π_0 is a degenerate distribution (or atom) at $\boldsymbol{\psi}^0$,

$$\pi_0(\boldsymbol{\psi}_n)=\delta_{\boldsymbol{\psi}^0}. \tag{7}$$

For $k = 1, \ldots, K$, assume that π_k has density (and we slightly abuse the notation π_k to also denote the density function). Thus, the prior density of $\boldsymbol{\psi}_n$ (marginalized over Z_n), denoted by $\pi(\boldsymbol{\psi}_n)$ is a mixture with an atom at $\boldsymbol{\psi}^0$, given by

$$\pi\left(\boldsymbol{\psi}_n\right) = q_0^{(n)} \delta_{\boldsymbol{\psi}^0} + \sum_{k=1}^{K} q_k^{(n)} \pi_k(\boldsymbol{\psi}_n). \tag{8}$$

P5. We assume the independence among $(\boldsymbol{\psi}_n)_{1:N}$, and denote them collectively as $\boldsymbol{\psi}$. Let $\pi(\boldsymbol{\psi})$ denote the prior density function of $\boldsymbol{\psi}$, which factorizes due to independence,

$$\pi(\boldsymbol{\psi}) = \prod_{n=1}^{N} \pi\left(\boldsymbol{\psi}_n\right). \tag{9}$$

The graphical structure of the proposed Bayesian model is shown in Figure 1 as a directed-acyclic-graph (DAG) using plate notations. This graphical illustration is helpful for visualizing the dependencies and conditional independence relations between parameters and random variables.

Our objective is to find an estimator $h(\boldsymbol{y})$ for $f_* := f(\mathbf{x}_*)$, the spatial field at location $\mathbf{x}_*$, based on observations $\boldsymbol{y} := (y_{n,m})_{n=1:N, m=1:M_n}$.

III. Posterior of the Bayesian Model

In this section we derive the following quantities of interest, based on which the Bayesian estimators will be developed in Sections IV and V:

1) The posterior distribution of the model parameters $\boldsymbol{\psi}$, given by $p(\boldsymbol{\psi}|\boldsymbol{y})$ (Theorem 1).
2) The posterior predictive distribution $p(f_*|\boldsymbol{y})$ (Theorem 2).

The following theorem gives the posterior density function of $\boldsymbol{\psi}$.

Theorem 1. *Let $\boldsymbol{\mu} := (\mu(\mathbf{x}_n))_{1:N} \in \mathbb{R}^N$ be the expected values of the F process at locations $(\mathbf{x}_n)_{1:N}$, and let $\boldsymbol{\mathcal{C}} \in \mathbb{R}^{N\times N}$ be the covariance matrix, where $(\boldsymbol{\mathcal{C}})_{ij} = \mathcal{C}(\mathbf{x}_i, \mathbf{x}_j)$. For $n = 1, \ldots, N$, define $g_n := \sum_{m=1}^{M_n} y_{n,m}$, $s_n := \sum_{m=1}^{M_n} y_{n,m}^2$. Let $\mathbf{M} := \mathrm{diag}[(M_n)_{1:N}]$, $\boldsymbol{a} := (a_n)_{1:N}$, $\mathbf{A} := \mathrm{diag}[\boldsymbol{a}]$, $\boldsymbol{b} := (b_n)_{1:N}$, $\boldsymbol{g} := (g_n)_{1:N}$, $\boldsymbol{s} := (s_n)_{1:N}$. Let $\tilde{g}_n := a_n^{-1}(M_n^{-1} g_n - b_n)$, $\tilde{\boldsymbol{g}} := (\tilde{g}_n)_{1:N} = \mathbf{A}^{-1}(\mathbf{M}^{-1}\boldsymbol{g} - \boldsymbol{b})$, $\boldsymbol{\Upsilon} := \boldsymbol{\mathcal{C}} + \varsigma^2 \mathbf{M}^{-1}$. Then, the log posterior density function of $\boldsymbol{\psi}$ is given by*

$$\begin{aligned}
&\log p(\boldsymbol{\psi}|\boldsymbol{y}) \\
=& -\frac{1}{2}\Big[\mathrm{tr}(\mathbf{M})\log 2\pi + \mathrm{tr}(\mathbf{M}\log(\varsigma^2\mathbf{A}^2)) - \log|\varsigma^2\mathbf{M}^{-1}| \\
&+ \log|\boldsymbol{\Upsilon}| + \varsigma^{-2}\mathbf{1}^T\mathbf{A}^{-2}\boldsymbol{s} - \varsigma^{-2}\boldsymbol{g}^T\mathbf{M}^{-1}\mathbf{A}^{-2}\boldsymbol{g} \\
&+ (\tilde{\boldsymbol{g}} - \boldsymbol{\mu})^T\boldsymbol{\Upsilon}^{-1}(\tilde{\boldsymbol{g}} - \boldsymbol{\mu})\Big] + \log\pi(\boldsymbol{\psi}) - \log p(\boldsymbol{y}).
\end{aligned} \tag{10}$$

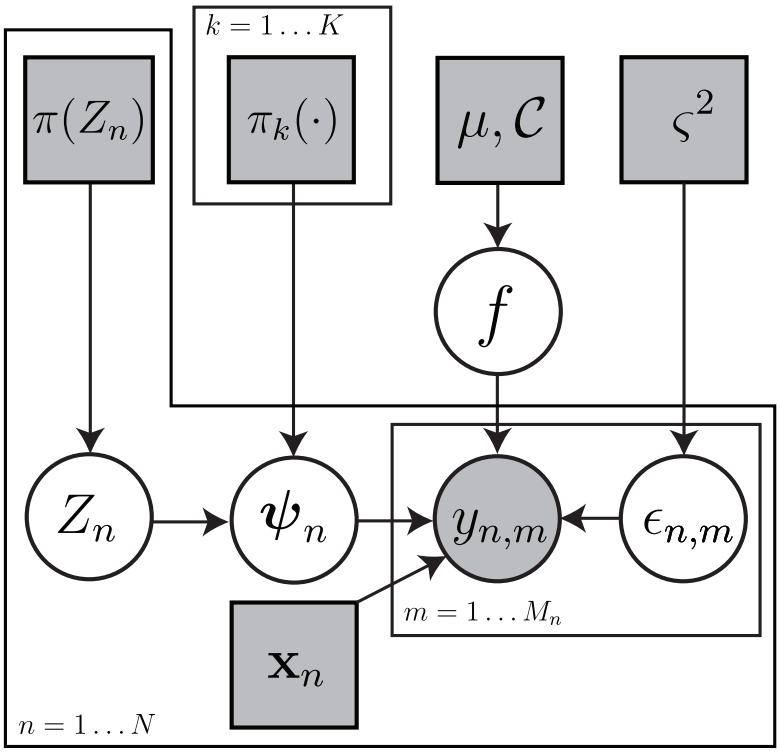


Fig. 1: Directed acyclic graph (DAG) of the model using the plate notation. Shaded rectangles represent constants and observed covariates. White circles represent unobserved random variables. The shaded circle represents observed random variable. Arrows represent conditional dependence between two quantities.

$p(\boldsymbol{y})$ is a normalizing constant that is analytically intractable.

Proof. See Appendix A-A. □

Remark 1. *As shown in the proof of Theorem 1, the statistics g_n and s_n are sufficient for $\boldsymbol{\psi}$. In fact, as we show later, the estimators depend on $\boldsymbol{y}$ only through $(M_n, g_n, s_n)_{n=1:N}$. Thus, from now on we take $\boldsymbol{y} := (M_n, g_n, s_n)_{n=1:N}$ as the summary of observations from all sensors.*

The following theorem gives the posterior predictive distribution of the model.

Theorem 2. *The posterior predictive density is given by*

$$p(f_*|\boldsymbol{y}) = \int p(f_*|\boldsymbol{y}, \boldsymbol{\psi}) p(\boldsymbol{\psi}|\boldsymbol{y}) d\boldsymbol{\psi}. \tag{11}$$

Let $\mu_* := \mu(\mathbf{x}_*)$, $\mathcal{C}_* := \mathcal{C}(\mathbf{x}_*, \mathbf{x}_*)$, *let* $\boldsymbol{k}_* := (\mathcal{C}(\mathbf{x}_n, \mathbf{x}_*))_{n=1:N} \in \mathbb{R}^N$ *be a column vector, and we have*

$$p(f_*|\boldsymbol{y}, \boldsymbol{\psi}) = \frac{1}{\sqrt{2\pi\sigma_*^2}} \exp\left(-\frac{(f_* - \bar{f}_*)^2}{2\sigma_*^2}\right), \tag{12}$$

$$\bar{f}_* = \mu_* + \boldsymbol{k}_*^T \boldsymbol{\Upsilon}^{-1}(\tilde{\boldsymbol{g}} - \boldsymbol{\mu}), \tag{13}$$

$$\sigma_*^2 = \mathcal{C}_* - \boldsymbol{k}_*^T \boldsymbol{\Upsilon}^{-1} \boldsymbol{k}_*. \tag{14}$$

Proof. See Appendix A-B. □

IV. Spatial Best Linear Unbiased Estimator (S-BLUE)

We now derive the Spatial Best Linear Unbiased Estimator (S-BLUE). Let $l(h(\boldsymbol{y}), f_*)$ denote the loss function, i.e. the loss incurred when using estimator h when in fact the target quantity is f_*. Let $R[\Pi, h]$ denote the Bayes risk of h associated with the prior distribution Π, which is defined as the expected value of loss taken over Π, i.e.

$$R[\Pi, h] = \mathbb{E}[l(h(\boldsymbol{y}), f_*)].$$

To derive the S-BLUE, we restrict the estimator to be a member of the family of linear estimators, that is $\mathcal{H} := \{h(\boldsymbol{y}) = \boldsymbol{w}^T \boldsymbol{y} + b\}$, where $\boldsymbol{w}$ is a weight vector and b is an intercept, both of which do not depend on $\boldsymbol{y}$ or any unknown variables. Hence, the S-BLUE is defined as the optimal linear estimator under quadratic loss, $l(h(\boldsymbol{y}), f_*) = (h(\boldsymbol{y}) - f_*)^2$, and is given by

$$\widehat{h}_{\text{S-BLUE}} = \arg\min_{h \in \mathcal{H}} R[\Pi, h] = \arg\min_{h \in \mathcal{H}} \mathbb{E}\left[(h(\boldsymbol{y}) - f_*)^2\right], \tag{15}$$

where the expectation is taken over the joint distribution of r.v.'s $(f_*, \boldsymbol{y}, \boldsymbol{\psi})$. The next theorem shows that $\widehat{h}_{\text{S-BLUE}}$ can be expressed in closed-form.

Theorem 3. $\widehat{h}_{\textit{S-BLUE}}$ *is given by*

$$\widehat{h}_{\textit{S-BLUE}}(\boldsymbol{y}) = \mu_* + \text{Cov}[\bar{\boldsymbol{g}}, f_*]^T \text{Cov}[\bar{\boldsymbol{g}}]^{-1}(\bar{\boldsymbol{g}} - \mathbb{E}[\bar{\boldsymbol{g}}]), \tag{16}$$

where $\bar{\boldsymbol{g}} = \mathbf{M}^{-1}\boldsymbol{g}$. $\mathbb{E}[\bar{\boldsymbol{g}}], \mathrm{Cov}[\bar{\boldsymbol{g}}, f_*], \mathrm{Cov}[\bar{\boldsymbol{g}}]$ *can all be expressed in closed-form (let* $\odot$ *denote matrix entry-wise multiplication),*

$$\mathbb{E}[\bar{\boldsymbol{g}}] = \mathrm{diag}\left(\mathbb{E}[\boldsymbol{a}]\right)\boldsymbol{\mu} + \mathbb{E}[\boldsymbol{b}], \tag{17}$$

$$\mathrm{Cov}[\bar{\boldsymbol{g}}, f_*] = \mathrm{diag}\left(\mathbb{E}[\boldsymbol{a}]\right)\boldsymbol{k}_*, \tag{18}$$

$$\begin{aligned}\mathrm{Cov}[\bar{\boldsymbol{g}}] =& \mathbb{E}[\boldsymbol{a}\boldsymbol{a}^T] \odot \left(\boldsymbol{\mathcal{C}} + \varsigma^2\mathbf{M}^{-1} + \boldsymbol{\mu}\boldsymbol{\mu}^T\right) \\ &+ \mathrm{diag}(\boldsymbol{\mu})\left(\mathbb{E}[\boldsymbol{a}\boldsymbol{b}^T] + \mathbb{E}[\boldsymbol{a}\boldsymbol{b}^T]^T\right) \\ &+ \mathbb{E}[\boldsymbol{b}\boldsymbol{b}^T] - \mathbb{E}[\bar{\boldsymbol{g}}]\mathbb{E}[\bar{\boldsymbol{g}}]^T.\end{aligned} \tag{19}$$

The various terms in the above equations can all be computed in closed-form, and the details are given in the proof.

Proof. See Appendix A-C. □

The next corollary shows the unbiasedness property of $\widehat{h}_{\text{S-BLUE}}$.

Corollary 1. $\widehat{h}_{\textit{S-BLUE}}$ *is unbiased, that is,* $\mathbb{E}[\widehat{h}_{\textit{S-BLUE}}(\boldsymbol{y})] = \mathbb{E}[f_*]$.

Proof. It is shown via the linearity of expectation. □

The following corollary gives the closed-form expression of $R[\Pi, \widehat{h}_{\text{S-BLUE}}]$ (under quadratic loss).

Corollary 2. *Under quadratic loss, the Bayes risk associated with* $\widehat{h}_{\textit{S-BLUE}}$ *is given by*

$$R[\Pi, \widehat{h}_{\textit{S-BLUE}}] = \mathcal{C}_* - \mathrm{Cov}[\bar{\boldsymbol{g}}, f_*]^T\mathrm{Cov}[\bar{\boldsymbol{g}}]^{-1}\mathrm{Cov}[\bar{\boldsymbol{g}}, f_*]. \tag{20}$$

Proof. Substituting (16) into $R[\Pi, h] = \mathbb{E}[l(h(\boldsymbol{y}), f_*)]$ gives the proof. □

The complete S-BLUE algorithm is shown in Algorithm 1. Notice that much of the computation in S-BLUE does not require $\boldsymbol{y}$, and thus can be performed in an "offline phase", e.g. when the sensor network is deployed. In Algorithm 1, only Line 6 needs to be computed when the sensor measurements are taken. Below is a line-by-line analysis of the computational time complexity of Algorithm 1.

- Offline phase:
  - Line 1: Evaluating $\mathbb{E}[\boldsymbol{a}], \mathbb{E}[\boldsymbol{b}]$ takes $\mathcal{O}(KN)$; evaluating $\mathbb{E}[\boldsymbol{a}\boldsymbol{a}^T], \mathbb{E}[\boldsymbol{b}\boldsymbol{b}^T], \mathbb{E}[\boldsymbol{a}\boldsymbol{b}^T]$ takes $\mathcal{O}(KN^2)$.
  - Line 2: Evaluating $\mu_*, \mathcal{C}_*$ takes $\mathcal{O}(1)$; evaluating $\boldsymbol{\mu}, \boldsymbol{k}_*$ takes $\mathcal{O}(N)$; evaluating $\boldsymbol{\mathcal{C}}$ takes $\mathcal{O}(N^2)$.
  - Line 3: Evaluating $\mathbb{E}[\bar{\boldsymbol{g}}], \mathrm{Cov}[\bar{\boldsymbol{g}}, f_*]$ takes $\mathcal{O}(N)$; evaluating $\mathrm{Cov}[\bar{\boldsymbol{g}}]$ takes $\mathcal{O}(N^2)$.

Algorithm 1: Spatial-Best Linear Unbiased Estimator (S-BLUE)

Input: $\mathbf{x}_*$, $(\mathbf{x}_n)_{n=1:N}$, $\boldsymbol{y}$, $\left(q_k^{(n)}\right)_{n=1:N,k=0:K}$, $(\pi_k)_{k=1:K}$

Output: Estimator $\widehat{h}_{\text{S-BLUE}}(\boldsymbol{y})$, Bayes risk $R[\Pi, \widehat{h}_{\text{S-BLUE}}]$

Offline phase

1 Compute $\mathbb{E}[\boldsymbol{a}], \mathbb{E}[\boldsymbol{b}], \mathbb{E}[\boldsymbol{a}\boldsymbol{a}^T], \mathbb{E}[\boldsymbol{b}\boldsymbol{b}^T], \mathbb{E}[\boldsymbol{a}\boldsymbol{b}^T]$ (see Appendix A-C).

2 Compute $\mu_*, \boldsymbol{\mu}, \boldsymbol{k}_*, \boldsymbol{\mathcal{C}}, \mathcal{C}_*$ by their respective definitions.

3 Compute $\mathbb{E}[\bar{\boldsymbol{g}}], \mathrm{Cov}[\bar{\boldsymbol{g}}, f_*], \mathrm{Cov}[\bar{\boldsymbol{g}}]$ by Equations (17) - (19).

4 Compute $\widehat{\boldsymbol{w}} = \mathrm{Cov}[\bar{\boldsymbol{g}}]^{-1}\mathrm{Cov}[\bar{\boldsymbol{g}}, f_*]$, $\widehat{b} = \mu_* - \widehat{\boldsymbol{w}}^T\mathbb{E}[\bar{\boldsymbol{g}}]$.

5 Compute $R[\Pi, \widehat{h}_{\text{S-BLUE}}]$ by (20).

Online phase

6 After collecting sensor measurements $\boldsymbol{y}$, compute $\widehat{h}_{\text{S-BLUE}}(\boldsymbol{y}) = \widehat{\boldsymbol{w}}^T\bar{\boldsymbol{g}} + \widehat{b}$.

7 **return** $\widehat{h}_{\text{S-BLUE}}(\boldsymbol{y})$, $R[\Pi, \widehat{h}_{\text{S-BLUE}}]$.

- Line 4: Evaluating $\widehat{\boldsymbol{w}}$ takes $\mathcal{O}(N^3)$, since it involves solving a linear system; evaluating $\widehat{b}$ takes $\mathcal{O}(N)$.
- Line 5: Evaluating $R[\Pi, \widehat{h}_{\text{S-BLUE}}]$ takes $\mathcal{O}(N)$, since $\mathrm{Cov}[\bar{\boldsymbol{g}}]^{-1}\mathrm{Cov}[\bar{\boldsymbol{g}}, f_*]$ has been computed in Line 4.

- Online phase:
  - Line 6: Evaluating $\widehat{h}_{\text{S-BLUE}}(\boldsymbol{y})$ takes $\mathcal{O}(N)$.

Overall, the offline phase of Algorithm 1 takes $\mathcal{O}(N^3)$ (assuming that $K \leq N$), and its online phase takes $\mathcal{O}(N)$. It is worth noting that there are techniques to further reduce the computational complexity of the matrix inversion, e.g. through low-rank approximation (see [30]).

V. Empirical Bayes Estimators

We now derive an algorithm in which we do not restrict the estimator to be linear. The idea of empirical Bayes is to plug in a point estimate $\hat{\boldsymbol{\psi}}$ into (12) to approximate the posterior predictive distribution, i.e. $p(\boldsymbol{\psi}|\boldsymbol{y}) \approx \delta_{\hat{\boldsymbol{\psi}}}$. This gives us the corresponding empirical Bayes estimators, which minimize the expected posterior loss, conditional on $\hat{\boldsymbol{\psi}}$:

$$\widehat{h}_{\text{EB}}(\boldsymbol{y}, \hat{\boldsymbol{\psi}}) = \arg\min_{h(\boldsymbol{y})} \mathbb{E}[l(h(\boldsymbol{y}), f_*)|\boldsymbol{y}, \hat{\boldsymbol{\psi}}]. \tag{21}$$

To complete the specification of the estimator we are required to define appropriate *loss functions*. We present a few widely used loss functions and their corresponding approximate Bayes estimators.

1) **Quadratic loss function:** $l_{\text{quad}}(h(\boldsymbol{y}), f_*) = (h(\boldsymbol{y}) - f_*)^2$.

The corresponding Bayes estimator is the conditional expectation (minimum mean squared error estimator, or MMSE estimator),

$$\widehat{h}_{\text{MMSE}}(\boldsymbol{y}) = \mathbb{E}[f_*|\boldsymbol{y}] = \int_{\mathbb{R}} f_* p(f_*|\boldsymbol{y}) df_*,$$

where $p(f_*|\boldsymbol{y})$ is given in Theorem 2. The empirical Bayes version of $\widehat{h}_{\text{MMSE}}(\boldsymbol{y})$ is given by

$$\begin{aligned}\widehat{h}_{\text{EB-MMSE}}(\boldsymbol{y}, \hat{\boldsymbol{\psi}}) =& \mathbb{E}[f_*|\boldsymbol{y}, \hat{\boldsymbol{\psi}}] = \int_{\mathbb{R}} f_* p(f_*|\boldsymbol{y}, \hat{\boldsymbol{\psi}}) df_* \\ \approx& \widehat{h}_{\text{MMSE}}(\boldsymbol{y}),\end{aligned}$$

where $p(f_*|\boldsymbol{y}, \hat{\boldsymbol{\psi}})$ is given in (12).

2) **Absolute loss function:** $l_{\text{abs}}(h(\boldsymbol{y}), f_*) = |h(\boldsymbol{y}) - f_*|$.

The corresponding Bayes estimator is the conditional median (least absolute deviation estimator, or LAD estimator),

$$\widehat{h}_{\text{LAD}}(\boldsymbol{y}) = \text{median}(f_*|\boldsymbol{y}).$$

The empirical Bayes version of $\widehat{h}_{\text{LAD}}(\boldsymbol{y})$ is given by

$$\widehat{h}_{\text{EB-LAD}}(\boldsymbol{y}, \hat{\boldsymbol{\psi}}) = \text{median}(f_*|\boldsymbol{y}, \hat{\boldsymbol{\psi}}) \approx \widehat{h}_{\text{LAD}}(\boldsymbol{y}).$$

3) $0-1$ **loss function:** $l_{\text{0-1}}(h(\boldsymbol{y}), f_*) = \mathbb{1}_{\{f_* < h(\boldsymbol{y}) \leq f_* + df_*\}}$.

The corresponding Bayes estimator is the conditional mode (maximum a posteriori estimator, or MAP estimator),

$$\widehat{h}_{\text{MAP}}(\boldsymbol{y}) = \arg\max_{f_*} p(f_*|\boldsymbol{y}).$$

The empirical Bayes version of $\widehat{h}_{\text{MAP}}(\boldsymbol{y})$ is given by

$$\widehat{h}_{\text{EB-MAP}}(\boldsymbol{y}, \hat{\boldsymbol{\psi}}) = \arg\max_{f_*} p(f_*|\boldsymbol{y}, \hat{\boldsymbol{\psi}}) \approx \widehat{h}_{\text{MAP}}(\boldsymbol{y}).$$

For all the aforementioned Bayes estimators, we first need to find a point estimator for $\boldsymbol{\psi}$. To achieve this, we find the MAP estimator of $\boldsymbol{\psi}$, which aims at maximizing the posterior density

$p(\boldsymbol{\psi}|\boldsymbol{y})$ given in Theorem 1. The MAP estimator is then given by

$$\begin{aligned}\hat{\boldsymbol{\psi}} &= \arg\max_{\boldsymbol{\psi}} p\left(\boldsymbol{\psi}|\boldsymbol{y}\right) \\ &= \arg\max_{\boldsymbol{\psi}} \Bigg[-\frac{1}{2}\left\{\mathrm{tr}(\mathbf{M})\log 2\pi + \mathrm{tr}(\mathbf{M}\log(\varsigma^2\mathbf{A}^2))\right. \\ &\quad -\log|\varsigma^2\mathbf{M}^{-1}| + \log|\boldsymbol{\Upsilon}| + \varsigma^{-2}\mathbf{1}^T\mathbf{A}^{-2}\boldsymbol{s} \\ &\quad \left.-\varsigma^{-2}\boldsymbol{g}^T\mathbf{M}^{-1}\mathbf{A}^{-2}\boldsymbol{g} + (\tilde{\boldsymbol{g}}-\boldsymbol{\mu})^T\boldsymbol{\Upsilon}^{-1}(\tilde{\boldsymbol{g}}-\boldsymbol{\mu})\right\} \\ &\quad + \log\pi(\boldsymbol{\psi})\Bigg].\end{aligned} \tag{22}$$

Note that the optimization objective does not involve $p(\boldsymbol{y})$, since it is a constant. Thus, the empirical Bayes estimators could be computed in the following two-stage algorithm:

1) Compute $\hat{\boldsymbol{\psi}}$ by solving the optimization problem $\arg\max_{\boldsymbol{\psi}} p\left(\boldsymbol{\psi}|\boldsymbol{y}\right)$.
2) Plug in $\hat{\boldsymbol{\psi}}$ to compute $\widehat{h}_{\mathrm{EB}}(\boldsymbol{y}, \hat{\boldsymbol{\psi}})$.

In order to solve the optimization problem in Step I, we develop two algorithms. The first approach is a stochastic optimization method named Cross-Entropy method (CEM), and the second approach is the Iterated Conditional Modes (ICM) which is based on iterative greedy search.

A. Cross-Entropy Method (CEM)

The Cross-Entropy method (CEM) is an stochastic algorithm that is suitable for solving combinatoric or continuous optimization problems. Suppose we have a maximization problem with a unique optimizer,

$$\hat{\boldsymbol{\varphi}} = \arg\max_{\boldsymbol{\varphi}\in\Phi} J(\boldsymbol{\varphi}),$$

where $J(\cdot)$ is the objective function, Φ is the domain, and $\boldsymbol{\varphi}$ is the parameter vector. We solve the optimization problem by considering the level sets of the objective function $\{\boldsymbol{\varphi} : J(\boldsymbol{\varphi}) \geq \gamma\}$, for $\gamma \in \mathbb{R}$. When $\gamma = \hat{J} = \max_{\boldsymbol{\varphi}\in\Phi} J(\boldsymbol{\varphi})$, we have $\{\boldsymbol{\varphi} : J(\boldsymbol{\varphi}) \geq \gamma\} = \{\hat{\boldsymbol{\varphi}}\}$. Next, let us define a family of probability measures $\{\mathbb{P}_{\boldsymbol{\theta}} : \boldsymbol{\theta} \in \boldsymbol{\Theta}\}$ on Φ with densities $\{w_{\boldsymbol{\theta}} : \boldsymbol{\theta} \in \boldsymbol{\Theta}\}$ that are parameterized by $\boldsymbol{\theta} \in \boldsymbol{\Theta}$. Let $\mathbb{E}_{\boldsymbol{\theta}}$ denote the expectation taken with respect to $\mathbb{P}_{\boldsymbol{\theta}}$. Let us fix $\boldsymbol{\theta}$ and γ, and define a rare event probability problem,

$$\mathbb{P}_{\boldsymbol{\theta}}[J(\boldsymbol{\varphi}) \geq \gamma] = \mathbb{E}_{\boldsymbol{\theta}}[\mathbb{1}_{\{J(\boldsymbol{\varphi})\geq\gamma\}}] = \int_{\Phi} \mathbb{1}_{\{J(\boldsymbol{\varphi})\geq\gamma\}} w_{\boldsymbol{\theta}}(\boldsymbol{\varphi}) d\boldsymbol{\varphi}.$$

Instead of approximating this probability naively by sampling from $w_{\boldsymbol{\theta}}$, the importance sampling method is used. Let $w_{\tilde{\boldsymbol{\theta}}}$ denote the importance sampler, where $\tilde{\boldsymbol{\theta}} \in \boldsymbol{\Theta}$. Importance sampling approximates the rare event probability by,

$$\begin{aligned}\mathbb{P}_{\boldsymbol{\theta}}[J(\boldsymbol{\varphi}) \geq \gamma] =& \int_{\Phi} \mathbb{1}_{\{J(\boldsymbol{\varphi})\geq\gamma\}} w_{\boldsymbol{\theta}}(\boldsymbol{\varphi}) d\boldsymbol{\varphi} \\ =& \mathbb{E}_{\tilde{\boldsymbol{\theta}}}\left[\mathbb{1}_{\{J(\boldsymbol{\varphi})\geq\gamma\}} \frac{w_{\boldsymbol{\theta}}(\boldsymbol{\varphi})}{w_{\tilde{\boldsymbol{\theta}}}(\boldsymbol{\varphi})}\right] \\ \approx& \frac{1}{S}\sum_{s=1}^{S} \mathbb{1}_{\{J(\tilde{\boldsymbol{\varphi}}^{[s]})\geq\gamma\}} \frac{w_{\boldsymbol{\theta}}(\tilde{\boldsymbol{\varphi}}^{[s]})}{w_{\tilde{\boldsymbol{\theta}}}(\tilde{\boldsymbol{\varphi}}^{[s]})},\end{aligned} \tag{23}$$

where $\tilde{\boldsymbol{\varphi}}^{[1]}, \ldots, \tilde{\boldsymbol{\varphi}}^{[S]}$ are S independent samples generated from $w_{\tilde{\boldsymbol{\theta}}}$. The optimal importance sampler $w_{\hat{\boldsymbol{\theta}}}$ is selected through the cross-entropy criterion,

$$\begin{aligned}\hat{\boldsymbol{\theta}} =& \arg\min_{\tilde{\boldsymbol{\theta}}\in\boldsymbol{\Theta}} \int_{\Phi} \mathbb{1}_{\{J(\boldsymbol{\varphi})\geq\gamma\}} w_{\boldsymbol{\theta}}(\boldsymbol{\varphi}) \log \frac{w_{\boldsymbol{\theta}}(\boldsymbol{\varphi})}{w_{\tilde{\boldsymbol{\theta}}}(\boldsymbol{\varphi})} d\boldsymbol{\varphi} \\ \approx& \arg\max_{\tilde{\boldsymbol{\theta}}\in\boldsymbol{\Theta}} \frac{1}{S}\sum_{s=1}^{S} \mathbb{1}_{\{J(\boldsymbol{\varphi}^{[s]})\geq\gamma\}} \log w_{\tilde{\boldsymbol{\theta}}}(\boldsymbol{\varphi}^{[s]}),\end{aligned} \tag{24}$$

where $\boldsymbol{\varphi}^{[1]}, \ldots, \boldsymbol{\varphi}^{[S]}$ are S independent samples generated from $w_{\boldsymbol{\theta}}$. Notice that the last line of (24) corresponds to the maximum likelihood estimation (MLE) of $\tilde{\boldsymbol{\theta}}$ when the samples are $\{\boldsymbol{\varphi}^{[s]} : J(\boldsymbol{\varphi}^{[s]}) \geq \gamma\}$. The CEM starts from an initial sampling distribution $w_{\hat{\boldsymbol{\theta}}_0}$ and iteratively updates the threshold $\hat{\gamma}$ and the sampling distribution $w_{\hat{\boldsymbol{\theta}}}$. For a detailed introduction of CEM, see [32]. The complete procedure is detailed in Algorithm 2.

We can now link CEM to the MAP estimation problem in (22). We define the objective function as the (un-normalized) log-posterior conditional density,

$$J(\boldsymbol{\psi}) = \log p\left(\boldsymbol{\psi}|\boldsymbol{y}\right) + \log \pi\left(\boldsymbol{\psi}\right).$$

For the purpose of demonstrating the CEM, let us assume here that under prior distribution π_k, $(\log a_n, b_n)$ have a bivariate normal distribution,

$$\begin{pmatrix} \log a_n \\ b_n \end{pmatrix} \sim \mathcal{N}(\boldsymbol{\nu}_k, \boldsymbol{\Xi}_k).$$

Note that this can be easily adapted for other prior distributions. We choose the family of sampling distributions such that,

$$\begin{aligned} w_{\boldsymbol{\theta}}(\boldsymbol{\psi}) =& \prod_{n=1}^{N} w_{\boldsymbol{\theta}^{(n)}}(\boldsymbol{\psi}_n), \\ w_{\boldsymbol{\theta}^{(n)}}(\boldsymbol{\psi}_n) =& r_0^{(n)} \delta_{\boldsymbol{\psi}^0} + \sum_{k=1}^{K} r_k^{(n)} \mathcal{N}((\log a_n, b_n)^T; \tilde{\boldsymbol{\nu}}_k^{(n)}, \tilde{\boldsymbol{\Xi}}_k^{(n)}). \end{aligned}$$

Algorithm 2: Cross-Entropy Method (CEM)-Based Optimizer

Input: number of importance samples S, $\rho \in (0,1)$ (typically $0.001 \leq \rho \leq 0.01$), initial sampler parameter $\hat{\boldsymbol{\theta}}_0$, objective function J

Output: $\hat{\boldsymbol{\varphi}} = \arg\max_{\boldsymbol{\varphi}\in\Phi} J(\boldsymbol{\varphi})$

1 $\hat{\gamma}_0 \leftarrow -\infty$, $t \leftarrow 1$.

2 **repeat**

3 Generate S independent samples $\boldsymbol{\varphi}^{[1]}, \ldots, \boldsymbol{\varphi}^{[S]}$ from $w_{\hat{\boldsymbol{\theta}}_{t-1}}$.

4 Compute $J\left(\boldsymbol{\varphi}^{[1]}\right), \ldots, J\left(\boldsymbol{\varphi}^{[S]}\right)$.

5 $\hat{\gamma}_t \leftarrow$ the $(1-\rho)$-sample quantile of $J\left(\boldsymbol{\varphi}^{[1]}\right), \ldots, J\left(\boldsymbol{\varphi}^{[S]}\right)$.

6 $\hat{\boldsymbol{\theta}}_t \leftarrow \arg\max_{\boldsymbol{\theta}\in\boldsymbol{\Theta}} \frac{1}{S}\sum_{s=1}^{S} \mathbb{1}_{\{J(\boldsymbol{\varphi}^{[s]})\geq\hat{\gamma}_t\}} \log w_{\boldsymbol{\theta}}\left(\boldsymbol{\varphi}^{[s]}\right)$.

7 $t \leftarrow t+1$.

until *termination condition is triggered*;

8 Set $\hat{\boldsymbol{\varphi}}$ to be the sample with the largest $J(\hat{\boldsymbol{\varphi}})$ so far.

9 **return** $\hat{\boldsymbol{\varphi}}$.

Here, we have $\boldsymbol{\theta} = (\boldsymbol{\theta}^{(n)})_{n=1:N} = ((r_k^{(n)})_{0:K}, (\tilde{\boldsymbol{\nu}}_k^{(n)})_{1:K}, (\widetilde{\boldsymbol{\Xi}}_k^{(n)})_{1:K})_{n=1:N}$. Before running the CEM algorithm, we set $\hat{\boldsymbol{\theta}}_0$ such that $w_{\hat{\boldsymbol{\theta}}_0}$ coincides with the prior distribution π. Under this setting, the optimization in Line 6 of Algorithm 2 corresponds to the MLE of $\boldsymbol{\theta}$, given independent samples $\{\boldsymbol{\psi}^{[s]} : J(\boldsymbol{\psi}^{[s]}) \geq \hat{\gamma}_t\}$. This decomposes into sub-problems

$$\hat{\boldsymbol{\theta}}^{(n)} = \arg\max_{\boldsymbol{\theta}^{(n)}} \sum_{s=1}^{S} \mathbb{1}_{\{J(\boldsymbol{\psi}^{[s]})\geq\hat{\gamma}_t\}} \log w_{\boldsymbol{\theta}^{(n)}}(\boldsymbol{\psi}_n^{[s]}).$$

Since $w_{\boldsymbol{\theta}^{(n)}}$ is a mixture distribution, the MLE does not admit a closed-form solution and we use the expectation-maximization (EM) algorithm. The EM algorithm is an iterative procedure that computes a local optimum of the likelihood function. For notational simplicity, we drop the superscripts and subscripts with n for now, and denote the samples used to obtain the MLE as $\boldsymbol{\psi}^{[1]}, \ldots, \boldsymbol{\psi}^{[S]}$. To apply the EM algorithm, let us first introduce the auxiliary variables. Let $z^{[s]} \in \{0, \ldots, K\}$ for $s = 1, \ldots, S$ be the discrete auxiliary variables, such that

$$\begin{aligned}
p(\boldsymbol{\psi}^{[s]}|z^{[s]} = 0; \boldsymbol{\theta}) =& \delta_{\boldsymbol{\psi}^0} \\
p(\boldsymbol{\psi}^{[s]}|z^{[s]} = k; \boldsymbol{\theta}) =& \mathcal{N}((\log a, b)^T; \tilde{\boldsymbol{\nu}}_k, \widetilde{\boldsymbol{\Xi}}_k), \\
& \text{for } k = 1, \ldots, K, \\
p(z^{[s]} = k; \boldsymbol{\theta}) =& r_k, \text{ for } k = 0, \ldots, K.
\end{aligned}$$

Algorithm 3: Expectation-Maximization (EM) Algorithm

Input: S samples $\boldsymbol{\psi}^{[1]}, \ldots, \boldsymbol{\psi}^{[S]}$, initial estimate $\hat{\boldsymbol{\theta}}_0$

Output: Estimated parameter $\hat{\boldsymbol{\theta}}$

1 $\hat{p} \leftarrow -\infty$, $t \leftarrow 0$.

2 **repeat**

3 **for** $s = 1 \ldots S$ **do**

4 Compute $p_k^{[s]} := p(z^{[s]} = k | \boldsymbol{\psi}^{[s]}; \hat{\boldsymbol{\theta}}_t)$, for $k = 0, \ldots, K$.

5 *(E-step)* Construct $Q(\boldsymbol{\theta}; \hat{\boldsymbol{\theta}}_t) = \sum_{s=1}^{S} \sum_{k=0}^{K} p_k^{[s]} p(\boldsymbol{\psi}^{[s]} | z^{[s]} = k; \boldsymbol{\theta})$.

6 *(M-step)* $\hat{\boldsymbol{\theta}}_{t+1} \leftarrow \arg\max_{\boldsymbol{\theta} \in \boldsymbol{\Theta}} Q(\boldsymbol{\theta}; \hat{\boldsymbol{\theta}}_t)$. This decomposes into $K+1$ weighted MLE problems.

7 $t \leftarrow t + 1$.

until *termination condition is triggered*;

8 $\hat{\boldsymbol{\theta}} \leftarrow \hat{\boldsymbol{\theta}}_t$.

9 **return** $\hat{\boldsymbol{\theta}}$.

This gives the marginal distributions $w_{\boldsymbol{\theta}}(\boldsymbol{\psi}^{[s]})$ above. The EM algorithm starts with an initial estimate $\hat{\boldsymbol{\theta}}_0$, and iteratively updates the estimated parameter through two steps. In the expectation step (E-step), a lower bound of the log-likelihood function is constructed by first computing the conditional distributions of the auxiliary variables given the estimate of the parameters in the t-th iteration

$$p(z^{[s]} = k | \boldsymbol{\psi}^{[s]}; \hat{\boldsymbol{\theta}}_t) = \frac{p(\boldsymbol{\psi}^{[s]} | z^{[s]} = k; \hat{\boldsymbol{\theta}}_t) p(z^{[s]} = k; \hat{\boldsymbol{\theta}}_t)}{\sum_{k'=0}^{K} p(\boldsymbol{\psi}^{[s]} | z^{[s]} = k'; \hat{\boldsymbol{\theta}}_t) p(z^{[s]} = k'; \hat{\boldsymbol{\theta}}_t)},$$

for $k = 0, \ldots, K$, and then computing the expected value of the log-likelihood function with respect to this conditional distribution, given by

$$Q(\boldsymbol{\theta}; \hat{\boldsymbol{\theta}}_t) = \sum_{s=1}^{S} \mathbb{E}_{(z^{[s]} | \boldsymbol{\psi}^{[s]}; \hat{\boldsymbol{\theta}}_t)} [\log w_{\boldsymbol{\theta}}(\boldsymbol{\psi}^{[s]}, z^{[s]})].$$

In the maximization step (M-step), the estimated parameters in the $(t+1)$-th iteration are computed by maximizing the lower bound $Q(\boldsymbol{\theta}; \hat{\boldsymbol{\theta}}_t)$, that is,

$$\hat{\boldsymbol{\theta}}_{t+1} = \arg\max_{\boldsymbol{\theta} \in \boldsymbol{\Theta}} Q(\boldsymbol{\theta}; \hat{\boldsymbol{\theta}}_t).$$

The algorithm is summarized in Algorithm 3. For details about the EM algorithm, see [33].

To analyse the computational time complexity of CEM, let us first assume that each weighted MLE problem in Line 6 of the EM algorithm (Algorithm 3) takes $\mathcal{O}(S)$. For example, this is the

case when the sampling distribution is a mixture of normal distributions. The computational time complexity of Algorithm 3 is analysed as follows:

- Line 3-4: Each iteration takes $\mathcal{O}(K)$, the complexity is $\mathcal{O}(SK)$.
- Line 5: No actual computation is performed.
- Line 6: Computation of $K+1$ weighted MLE takes $\mathcal{O}(SK)$.

Thus, the total complexity of the EM algorithm is $T_{\text{EM}} = \mathcal{O}(J_{\text{EM}}SK)$, where J_{EM} is the number of iterations, which is usually quite small in practice. With this, we analyse the computational time complexity of the CEM estimator as follows:

- Preparation:
  - Evaluation of $\log|\boldsymbol{\Upsilon}|$ and $\boldsymbol{\Upsilon}^{-1}$ takes $\mathcal{O}(N^3)$.
- Algorithm 2:
  - Line 3: Generation of S samples takes $\mathcal{O}(SK)$ due to $w_{\hat{\boldsymbol{\theta}}_{t-1}}$ being a $(K+1)$-mixture.
  - Line 4: Evaluation of the objective function (22) S times takes $\mathcal{O}(SN^2)$.
  - Line 5: Computation of the sample quantile takes $\mathcal{O}(S)$.
  - Line 6: Computation of the maximizer using Algorithm 3 takes T_{EM}.
- Inference:
  - Evaluation of $\widehat{h}_{\text{EB}}(\boldsymbol{y}, \hat{\boldsymbol{\psi}})$ by (13), (14) takes $\mathcal{O}(N^2)$.

The total complexity of the CEM estimator is $\mathcal{O}(N^3 + J_{\text{CEM}}(SN^2 + T_{\text{EM}}))$, where J_{CEM} is the number of iterations in Algorithm 2. Since the EM algorithm converges rather quickly in practice, the complexity of CEM estimator is $\mathcal{O}(N^3 + J_{\text{CEM}}SN^2)$. It is worth noting that the preparation phase of the CEM estimation procedure can be run "offline", i.e. before having access to sensor measurements. Hence, the CEM estimator has complexity $\mathcal{O}(N^3)$ in the offline phase, and $\mathcal{O}(J_{\text{CEM}}SN^2)$ in the online phase. The same remark on the computational complexity of matrix inversion applies here as above.

B. Iterated Conditional Modes (ICM)

We propose a second optimization method to find the MAP estimator $\hat{\boldsymbol{\psi}}$ which is based on iterative greedy search. Since $\hat{\boldsymbol{\psi}} \in \mathbb{R}^{2N}$, the dimensionality of the optimization problem is high if N is large. In addition, for $n = 1, \ldots, N$, the distribution of $\boldsymbol{\psi}_n$ contains an atom. Hence, to improve the computational efficiency in these settings, we seek to reduce the complexity of the MAP estimation by reducing the global search problem to a sequence of iterative local search problems of iterated conditional modes (ICM) [34]. Let $\boldsymbol{\psi}_{(-n)} := (\boldsymbol{\psi}_m)_{m \neq n}$. In each iteration

of ICM, we fix $\boldsymbol{\psi}_{(-n)}$ and compute the mode of the conditional posterior distribution $\hat{\boldsymbol{\psi}}_n = \arg\max_{\boldsymbol{\psi}_n} p\left(\boldsymbol{\psi}_n|\boldsymbol{y}, \boldsymbol{\psi}_{(-n)}\right)$ through the conjugate gradient algorithm. ICM converges to a local maximum of the objective function in the sense that $\hat{\boldsymbol{\psi}}_n$ is the mode of the conditional posterior distribution for $n = 1, \ldots, N$.

To optimize the conditional posterior distributions, we decomposed them (up to a normalizing constant) as follows, for $n = 1, \ldots, N$,

$$\begin{aligned} p\left(\boldsymbol{\psi}_n|\boldsymbol{y}, \boldsymbol{\psi}_{(-n)}\right) \propto & p\left(\boldsymbol{y}|\boldsymbol{\psi}\right) \pi\left(\boldsymbol{\psi}_n\right) \\ \propto & p\left(\boldsymbol{y}_n|\boldsymbol{y}_{(-n)}, \boldsymbol{\psi}\right) \pi\left(\boldsymbol{\psi}_n\right), \end{aligned}$$

where $\boldsymbol{y}_n = (y_{n,m})_{m=1:M_n}$, $\boldsymbol{y}_{(-n)} = (y_{i,m})_{i\neq n, m=1:M_i}$. Let $\mu_n := \mu(\mathbf{x}_n)$, $\boldsymbol{\mu}_{(-n)} := (\mu_i)_{i\neq n}$. Let $\mathcal{C}_n := \mathcal{C}(\mathbf{x}_n, \mathbf{x}_n)$. Let $\boldsymbol{\Upsilon}_{(-n,n)} \in \mathbb{R}^{(N-1)}$ denote the sub-matrix of $\boldsymbol{\Upsilon}$ involving the cross-terms between sensor n and the rest of sensors. Let $\boldsymbol{\Upsilon}_{(-n)} \in \mathbb{R}^{(N-1)\times(N-1)}$ denote the sub-matrix of $\boldsymbol{\Upsilon}$ related to sensors other than n. Let $\tilde{\boldsymbol{g}}_{(-n)} := (\tilde{g}_i)_{i\neq n}$. Completely analogous to Theorem 2, we have that,

$$\left(F(\mathbf{x}_n)|\boldsymbol{y}_{(-n)}, \boldsymbol{\psi}\right) \sim \mathcal{N}\left(\nu_n, \zeta_n\right),$$

where

$$\nu_n = \mu_n + \boldsymbol{\Upsilon}_{(-n,n)}{}^T \boldsymbol{\Upsilon}_{(-n)}{}^{-1} \left(\tilde{\boldsymbol{g}}_{(-n)} - \boldsymbol{\mu}_{(-n)}\right), \tag{25}$$

$$\zeta_n = \mathcal{C}_n + \varsigma^2 - \boldsymbol{\Upsilon}_{(-n,n)}{}^T \boldsymbol{\Upsilon}_{(-n)}{}^{-1} \boldsymbol{\Upsilon}_{(-n,n)}. \tag{26}$$

One verifies that ν_n and ζ_n do not depend on $\boldsymbol{\psi}_n$. Therefore, following a derivation similar to that in Theorem 1, we have,

$$\begin{aligned} & \log p\left(\boldsymbol{y}_n|\boldsymbol{y}_{(-n)}, \boldsymbol{\psi}\right) \\ = & \log\left[\int p(\boldsymbol{y}_n|f(\mathbf{x}_n), \boldsymbol{\psi}) p(f(\mathbf{x}_n)|\boldsymbol{y}_{(-n)}, \boldsymbol{\psi}) df(\mathbf{x}_n)\right] \\ = & -\frac{1}{2}\left[M_n \log 2\pi + (M_n - 1)\log(\varsigma^2 a_n^2)\right. \\ & + \log(a_n^2 M_n \zeta_n + \varsigma^2 a_n^2) + \varsigma^{-2} a_n^{-2}(s_n - M_n^{-1} g_n^2) \\ & \left. + (\zeta_n + \varsigma^2 M_n^{-1})^{-1}(\tilde{g}_n - \nu_n)^2\right]. \end{aligned} \tag{27}$$

Thus, the log-conditional likelihood as well as its partial derivatives can be efficiently evaluated. The ICM algorithm then separately treats the continuous and discrete parts of the parameter space, that is, comparing $\sup_{\boldsymbol{\psi}_n \neq \boldsymbol{\psi}^0} \log p\left(\boldsymbol{\psi}_n|\boldsymbol{y}, \boldsymbol{\psi}_{(-n)}\right)$ and $\log p\left(\boldsymbol{\psi}^0|\boldsymbol{y}, \boldsymbol{\psi}_{(-n)}\right)$.

The details of the ICM algorithm are shown in Algorithm 4. The computational time complexity of the ICM estimator is analysed as follows:

- Preparation:
  - Evaluation of $\boldsymbol{\Upsilon}^{-1}$ takes $\mathcal{O}(N^3)$.
- Algorithm 4:
  - Line 2-3: Evaluation of ν_n, ζ_n for $n = 1, \ldots, N$ takes $\mathcal{O}(N^3)$. Notice that the computation of ${\boldsymbol{\Upsilon}_{(-n)}}^{-1}\boldsymbol{\Upsilon}_{(-n,n)}$ can be simplified to $\mathcal{O}(N^2)$ via block-wise inversion once $\boldsymbol{\Upsilon}^{-1}$ has been computed.
  - Line 6: Assume that the 2-dimensional optimization takes $\mathcal{O}(T_{\text{CG}})$.
  - Line 7-9: The complexity is $\mathcal{O}(1)$.
- Inference:
  - Evaluation of $\widehat{h}_{\text{EB}}(\boldsymbol{y}, \hat{\boldsymbol{\psi}})$ by (13), (14) takes $\mathcal{O}(N^2)$.

The total complexity of the ICM estimator is thus $\mathcal{O}(N^3 + J_{\text{ICM}} N T_{\text{CG}})$, where J_{ICM} is the number of iterations in Algorithm 4. Similar to CEM, the preparation phase of ICM and Line 2-3 of Algorithm 4 can also be run offline. Hence, the ICM estimator has complexity $\mathcal{O}(N^3)$ in the offline phase, and $\mathcal{O}(N^2 + J_{\text{ICM}} N T_{\text{CG}})$ in the online phase. The same remark on the computational complexity of matrix inversion applies here as above.

To account for the multi-modality of the posterior distribution, we adopt a standard multiple start initialization strategy, that is to run ICM from a number of random initial estimates. This corresponds to running Algorithm 4 multiple times with different initial values.

VI. Distributed Approaches

Now, let us consider a large scale sensor network with $I \geq 2$ clusters of sensors, where each cluster has a cluster head that locally aggregates data to be sent to the global Fusion Center. The clusters are assumed to be disjoint. We reconstruct the spatial field in a distributed manner. For a spatial location $\mathbf{x}_* \in \mathcal{X}$, the estimation of $f(\mathbf{x}_*)$ is done in two steps:

1) Sensors within a cluster transmit the measurements to the cluster head (CH), and the CH performs a local estimation of $f(\mathbf{x}_*)$.
2) The I CHs transmit their local estimations to the FC, where local estimations are fused into the global estimation.

In this section, we develop fusion algorithms based on local S-BLUE and local empirical Bayes estimators.

Algorithm 4: Iterative Conditional Modes (ICM) Algorithm

Input: $(\mathbf{x}_n)_{n=1:N}$, $\boldsymbol{y}$, $\left(q_k^{(n)}\right)_{n=1:N,k=0:K}$, $(\pi_k)_{k=1:K}$

Output: Estimation of posterior mode $\hat{\boldsymbol{\psi}}$

Randomly initialize $\hat{\boldsymbol{\psi}}$.

for $n = 1 \dots N$ **do**

  Compute ν_n, ζ_n from (25) and (26).

repeat

  for $n = 1 \dots N$ **do**

    $\tilde{\boldsymbol{\psi}}_n \leftarrow \arg\max_{\boldsymbol{\psi}_n \in \mathbb{R}_+ \times \mathbb{R}, \boldsymbol{\psi}_n \neq \boldsymbol{\psi}^0} \log p\left(\boldsymbol{\psi}_n | \boldsymbol{y}, \hat{\boldsymbol{\psi}}_{(-n)}\right)$, by running the conjugate gradient algorithm disregarding the atom at $\boldsymbol{\psi}^0$.

    if $\log p\left(\boldsymbol{\psi}^0 | \boldsymbol{y}, \hat{\boldsymbol{\psi}}_{(-n)}\right) < \log p\left(\tilde{\boldsymbol{\psi}}_n | \boldsymbol{y}, \hat{\boldsymbol{\psi}}_{(-n)}\right)$ **then**

      $\hat{\boldsymbol{\psi}}_n \leftarrow \tilde{\boldsymbol{\psi}}_n$.

    else

      $\hat{\boldsymbol{\psi}}_n \leftarrow \boldsymbol{\psi}^0$.

until *termination condition is triggered*;

return $\hat{\boldsymbol{\psi}}$.

A. Distributed S-BLUE

Suppose that each cluster head i produces the local S-BLUE $\widehat{h}^{(i)}_{\text{S-BLUE}}(\boldsymbol{y}^{(i)})$ and its Bayes risk $R[\Pi, \widehat{h}^{(i)}_{\text{S-BLUE}}]$, where $\boldsymbol{y}^{(i)}$ denotes the sensor measurements collected from cluster i. The goal is to derive a rule to fuse $\{\widehat{h}^{(i)}_{\text{S-BLUE}}(\boldsymbol{y}^{(i)})\}_{i=1:I}$ into a single estimator $\widehat{h}_{\text{DS-BLUE}}(\boldsymbol{y})$, where the "D" in DS-BLUE stands for "distributed". We restrict ourselves by considering $\widehat{h}_{\text{DS-BLUE}}(\boldsymbol{y})$ as a convex combination of $\{\widehat{h}^{(i)}_{\text{S-BLUE}}(\boldsymbol{y}^{(i)})\}_{i=1:I}$, that is, $\widehat{h}_{\text{DS-BLUE}}(\boldsymbol{y}) := \sum_{i=1}^{I} c_i \widehat{h}^{(i)}_{\text{S-BLUE}}(\boldsymbol{y}^{(i)})$, where $c_i \geq 0$ for $i = 1, \dots, I$, and $\sum_{i=1}^{I} c_i = 1$ are the constraints required to preserve the unbiasedness of

DS-BLUE. The Bayes risk of $\widehat{h}_{\text{DS-BLUE}}$ is given by,

$$\begin{aligned}
&R[\Pi, \widehat{h}_{\text{DS-BLUE}}] \\
&= \sum_{i=1}^{I}\sum_{j=1}^{I} c_i c_j \mathbb{E}\left[\left(\widehat{h}^{(i)}_{\text{S-BLUE}}(\boldsymbol{y}^{(i)}) - f_*\right)\left(\widehat{h}^{(j)}_{\text{S-BLUE}}(\boldsymbol{y}^{(j)}) - f_*\right)\right] \\
&\leq \sum_{i=1}^{I}\sum_{j=1}^{I} c_i c_j \sqrt{R[\Pi, \widehat{h}^{(i)}_{\text{S-BLUE}}]}\sqrt{R[\Pi, \widehat{h}^{(j)}_{\text{S-BLUE}}]} \\
&= \left(\sum_{i=1}^{I} c_i \sqrt{R[\Pi, \widehat{h}^{(i)}_{\text{S-BLUE}}]}\right)^2 := \overline{R}[\Pi, \widehat{h}_{\text{DS-BLUE}}],
\end{aligned}$$

where the inequality in the third line above is by the Cauchy-Schwarz inequality. The coefficients $(c_i)_{1:I}$ are chosen to minimize the upper bound $\overline{R}[\Pi, \widehat{h}_{\text{DS-BLUE}}]$, which gives the following optimal values,

$$c_i^\star = \begin{cases} 1 & \text{if } i = \arg\min_{1\leq j\leq I} R[\Pi, \widehat{h}^{(j)}_{\text{S-BLUE}}], \\ 0 & \text{otherwise.} \end{cases} \tag{28}$$

It is assumed above that there is no tie among $(R[\Pi, \widehat{h}^{(j)}_{\text{S-BLUE}}])_{j=1:I}$. If there is one, the tie can be broken arbitrarily. The DS-BLUE is defined via the optimal coefficients in (28), $\widehat{h}_{\text{DS-BLUE}}(\boldsymbol{y}) := \sum_{i=1}^{I} c_i^\star \widehat{h}^{(i)}_{\text{S-BLUE}}(\boldsymbol{y}^{(i)})$.

The complete DS-BLUE algorithm is shown in Algorithm 5. To analyse the computational complexity of Algorithm 5, assume for now that each cluster contains at most N_c sensor nodes. The line-by-line analysis of its computational time complexity is as follows:

- Offline phase:
  - Cluster head:
    - Line 2: The complexity is $\mathcal{O}(N_c^3)$, same as the offline phase of Algorithm 1.
  - Fusion center:
    - Line 3: Evaluation of $(c_i^\star)_{1:I}$ takes $\mathcal{O}(I)$.
- Online phase:
  - Cluster head:
    - Line 6: Evaluating $\widehat{h}^{(i)}_{\text{S-BLUE}}(\boldsymbol{y}^{(i)})$ takes $\mathcal{O}(N_c)$.
  - Fusion center:
    - Line 7: Evaluating $\widehat{h}_{\text{DS-BLUE}}(\boldsymbol{y})$ takes $\mathcal{O}(I)$.

Overall, for cluster heads, the offline phase takes $\mathcal{O}(N_c^3)$, and the online phase takes $\mathcal{O}(N_c)$. For the fusion center, both the offline phase and the online phase take $\mathcal{O}(I)$.

Algorithm 5: Distributed Spatial-Best Linear Unbiased Estimator (DS-BLUE)

Input: $\mathbf{x}_*$, $(\mathbf{x}_n)_{n=1:N}$, $\boldsymbol{y}$, $\left(q_k^{(n)}\right)_{n=1:N,k=0:K}$, $(\pi_k)_{k=1:K}$

Output: Estimator $\widehat{h}_{\text{DS-BLUE}}(\boldsymbol{y})$, upper bound on the Bayes risk $\overline{R}[\Pi, \widehat{h}_{\text{DS-BLUE}}]$

Offline phase

for *each cluster head* i *in parallel* **do**

  Compute $\widehat{\boldsymbol{w}}^{(i)}$, $\widehat{b}^{(i)}$, $R[\Pi, \widehat{h}^{(i)}_{\text{S-BLUE}}]$ as in Algorithm 1 and transmit $R[\Pi, \widehat{h}^{(i)}_{\text{S-BLUE}}]$ to the fusion center.

The fusion center computes the optimal coefficients $(c_i^\star)_{1:I}$ by (28) and the upper bound on the Bayes risk $\overline{R}[\Pi, \widehat{h}_{\text{DS-BLUE}}]$.

Online phase

for *each cluster head* i *in parallel* **do**

  Collect measurements $\boldsymbol{y}^{(i)}$ from sensors within the cluster i.

  Compute local S-BLUE $\widehat{h}^{(i)}_{\text{S-BLUE}}(\boldsymbol{y}^{(i)}) = \widehat{\boldsymbol{w}}^{(i)T}\bar{\boldsymbol{g}}^{(i)} + \widehat{b}^{(i)}$ and transmit to the fusion center.

The fusion center computes $\widehat{h}_{\text{DS-BLUE}}(\boldsymbol{y}) = \sum_{i=1}^{I} c_i^\star \widehat{h}^{(i)}_{\text{S-BLUE}}(\boldsymbol{y}^{(i)})$.

return $\widehat{h}_{\text{DS-BLUE}}(\boldsymbol{y})$, $\overline{R}[\Pi, \widehat{h}_{\text{DS-BLUE}}]$.

B. Distributed Empirical Bayes Estimator

Similar to DS-BLUE, suppose that each cluster head i computes an approximate posterior distribution $p(f_*|\boldsymbol{y}^{(i)}, \hat{\boldsymbol{\psi}}^{(i)})$, where $\boldsymbol{y}^{(i)}$ denotes the sensor measurements collected from cluster i and $\hat{\boldsymbol{\psi}}^{(i)}$ is a point estimate of the distortion parameters of sensors in cluster i. Let $\widehat{h}^{(i)}_{\text{EB-MMSE}}$ denote the empirical Bayes MMSE estimator produced by cluster head i. Same as DS-BLUE, let $\widehat{h}_{\text{DEB-MMSE}}(\boldsymbol{y}) := \sum_{i=1}^{I} c_i \widehat{h}^{(i)}_{\text{EB-MMSE}}(\boldsymbol{y}^{(i)})$ be a convex combination of the local estimators, where $c_i \geq 0$ for $i = 1, \ldots, I$, and $\sum_{i=1}^{I} c_i = 1$. We choose the fusion rule to be similar to $\widehat{h}_{\text{DS-BLUE}}$, that is,

$$c_i^\star = \begin{cases} 1 & \text{if } i = \arg\min_{1 \leq j \leq I} \operatorname{Var}[f_*|\boldsymbol{y}^{(j)}, \hat{\boldsymbol{\psi}}^{(j)}], \\ 0 & \text{otherwise,} \end{cases} \tag{29}$$

where ties are broken arbitrarily.

The complete distributed empirical Bayes algorithm is shown in Algorithm 6. For the cluster heads, the computational time complexity of this algorithm is the same as in the non-distributed version, with N replaced by N_c. For the fusion center, the complexity is $\mathcal{O}(I)$.

Algorithm 6: Distributed Empirical Bayes MMSE Estimator (DEB-MMSE)

Input: $\mathbf{x}_*$, $(\mathbf{x}_n)_{n=1:N}$, $\boldsymbol{y}$, $\left(q_k^{(n)}\right)_{n=1:N,k=0:K}$, $(\pi_k)_{k=1:K}$

Output: Estimator $\widehat{h}_{\text{DEB-MMSE}}(\boldsymbol{y})$

for *each cluster head* i *in parallel* **do**

  Compute $p(f_*|\boldsymbol{y}^{(i)}, \hat{\boldsymbol{\psi}}^{(i)})$ via either CEM or ICM.

  Compute $\widehat{h}^{(i)}_{\text{EB-MMSE}}(\boldsymbol{y}^{(i)})$, $\text{Var}\left[f_*|\boldsymbol{y}^{(i)}, \hat{\boldsymbol{\psi}}^{(i)}\right]$ and transmit to the fusion center.

The fusion center computes the optimal coefficients $(c_i^\star)_{1:I}$ by (29).

The fusion center computes $\widehat{h}_{\text{DEB-MMSE}}(\boldsymbol{y}) = \sum_{i=1}^I c_i^\star \widehat{h}^{(i)}_{\text{EB-MMSE}}(\boldsymbol{y}^{(i)})$.

return $\widehat{h}_{\text{DEB-MMSE}}(\boldsymbol{y})$.

Remark 2. *One downside of* $\widehat{h}_{\text{DS-BLUE}}(\boldsymbol{y})$ *and* $\widehat{h}_{\text{DEB-MMSE}}(\boldsymbol{y})$ *is that the reconstructed spatial field is discontinuous in space. There exists various techniques to avoid discontinuities by smoothing the reconstruction around the discontinuous boundary. However, these are left to future work.*

VII. Experiments with Synthetic Data

We conduct two experiments with synthetically generated data to study the performance of the methods we proposed including S-BLUE, CEM, and ICM. In Section VII-A, we study the sensitivity of the proposed methods to the strength of distortions. In Section VII-B, we perform a realistic simulation and analyse the overall performance of the proposed methods. In the studies, the two empirical Bayes-based estimators (CEM and ICM) use the quadratic loss function and hence correspond to $\widehat{h}_{\text{EB-MMSE}}$. The proposed methods are compared to two baselines, the "oracle" case in which the distortions $\boldsymbol{\psi}$ are known exactly, and the "naive" case in which distortions are disregarded in the prediction, i.e. $\hat{\boldsymbol{\psi}}_n = \boldsymbol{\psi}^0$, for $n = 1, \ldots, N$.

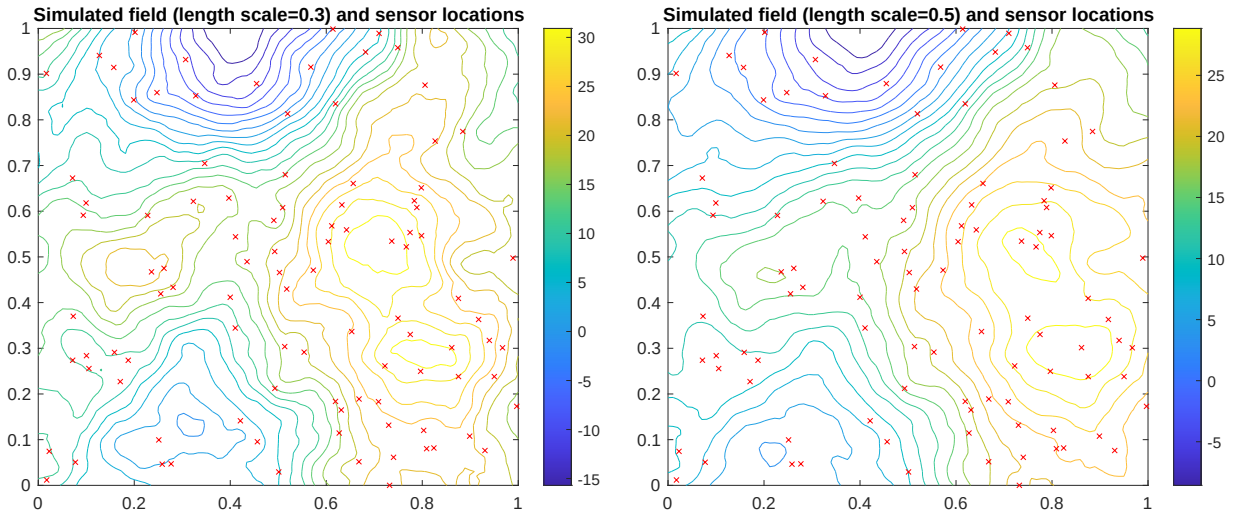
A. Synthetic Experiment 1: Homogeneous Distortion Characteristics

In this experiment, we study an ideal scenario where the signal-to-noise ratio (SNR) is high and observations are plentiful. We fix the number of observations per sensor to be 50, and the SNR to be 15dB. Notice that for the ease of comparison, all SNRs are measured at the sensor level, that is, the SNRs of aggregated observations. Under the i.i.d. noise assumption, we define $\text{SNR} = 10\log_{10}\left(\frac{M\text{Var}(F)}{\varsigma^2}\right)$, where $\text{Var}(F)$ denotes the signal variance, ς^2 denotes observation noise variance, and M denotes the number of observations per sensor.

We simulate a spatial field defined on the two-dimensional square $\mathcal{X} = [0,1]^2$ with mean 10, i.e. $\forall \mathbf{x} \in \mathcal{X}$, $\mu(\mathbf{x}) = 10$, and a Matérn covariance function with $\nu = 3/2$, $\mathrm{Var}(F) = 100$, and length scale=0.3, i.e. $\forall \mathbf{x}, \mathbf{x}' \in \mathcal{X}$, $\mathcal{C}(\mathbf{x}, \mathbf{x}') = 100 \left(1 + \frac{\sqrt{3}\|\mathbf{x}-\mathbf{x}'\|}{0.3}\right) \exp\left(-\frac{\sqrt{3}\|\mathbf{x}-\mathbf{x}'\|}{0.3}\right)$. Here $\|\mathbf{x} - \mathbf{x}'\|$ corresponds to the Euclidean distance. The contour plot of the simulated spatial field is shown in Figure 2a.

Subsequently, 100 sensors are randomly placed in the square. 50 out of the 100 sensors are fixed to have identical distortion parameters, and the rest are set to have the default transformation parameters $\boldsymbol{\psi}^0$, that is, non-distorting. For the sensors with distortions, we first fix the offset parameter b_n at 5, and vary the gain parameter a_n from 1 to 1.6. Then we fix the gain parameter at 1.2, and vary the offset parameter from 0 to 12. With each setting of distortion parameters, 100 sets of noisy observations are randomly simulated. For each set of observations, the three proposed methods: S-BLUE, CEM, and ICM, along with the two baselines oracle and naive, are used to reconstruct the spatial field. Here, weakly informative prior for the distortion parameters is used, which has a single category ($K = 1$) given by $q_1^{(n)} = 0.5$ for all n, where under π_1, $a_n \sim \log \mathcal{N}(0.25, 0.1^2)$, $b_n \sim \mathcal{N}(6, 3^2)$. The reconstruction accuracy is evaluated by the mean-squared-errors (MSE) at a 100×100 grid on $[0,1]^2$.

Figure 3 shows the reconstruction accuracy averaged over 100 realizations. For better interpretability, the ratio between the MSE and the prior variance, referred to as the relative MSE, is shown. Error bars in Figure 3 indicate the 95% Student's t-confidence interval of the relative MSE estimated from the 100 realizations. Error bars in all subsequent figures indicate the 95% Student's t-confidence interval of the respective underlying quantity. From Figure 3, one observes that the MSE of the oracle stayed constant, as expected, while the MSE of the naive baseline increased rapidly when the distortion parameters increased. The MSE of S-BLUE first decreased and then increased slightly. This is due to the way the prior distribution of distortions were set up. The prior mean of a_n and b_n were 1.29 and 6.0, respectively. Since S-BLUE makes predictions based purely on the prior information, its performance is best when the actual distortion is closest to the prior mean. The two empirical Bayes-based methods showed decreasing MSE when the gain parameter increased and slightly increasing MSE when the offset parameter increased. The reason is that since the gain parameter affects both the location and the spread of the observations, while the offset parameter only affects the location, the gain in the distortion was more noticeable and thus easier to detect. The MSEs of CEM and ICM were almost identical. Figure 4 shows the average false positive rate (FPR) and false negative rate



(a) Experiment 1 (b) Experiment 2

Fig. 2: Contour plots of simulated spatial fields used in the two synthetic experiments with sensor locations.

(FNR) of CEM and ICM. The FPR is defined as the proportion of non-distorting sensors that were estimated to be distorting, and the FNR is defined as the proportion of distorting sensors that were estimated to be non-distorting. The FPR and FNR of the two methods were almost identical. Notice in addition that the FNR was high when the gain was 1 and the offset was 5. This was caused by the short length scale (0.3), which made detection of the offset hard due to the low spatial correlation. Finally, all error bars are narrow, indicating that difference between the performance of different methods are statistically significant. To further confirm this, we show the maximum absolute deviation from the relative MSE of the five methods in the second column of Table I. One checks that the deviations are small, indicating that the performance is stable across realizations.

B. Synthetic Experiment 2: Inhomogeneous Distortion Characteristics

In the second synthetic experiment, we study a realistic scenario where each sensor has different distortion parameters, and vary the SNR as well as the number of observations.

We again simulate a spatial field defined on the two-dimensional square $[0, 1]^2$. This time, however, the length scale is set to be 0.5, and the spatial correlation decays at a slower rate. The contour plot of the simulated spatial field is shown in Figure 2b. The 100 sensors are placed at the same locations as in the synthetic experiment 1. 50 out of the 100 sensors are

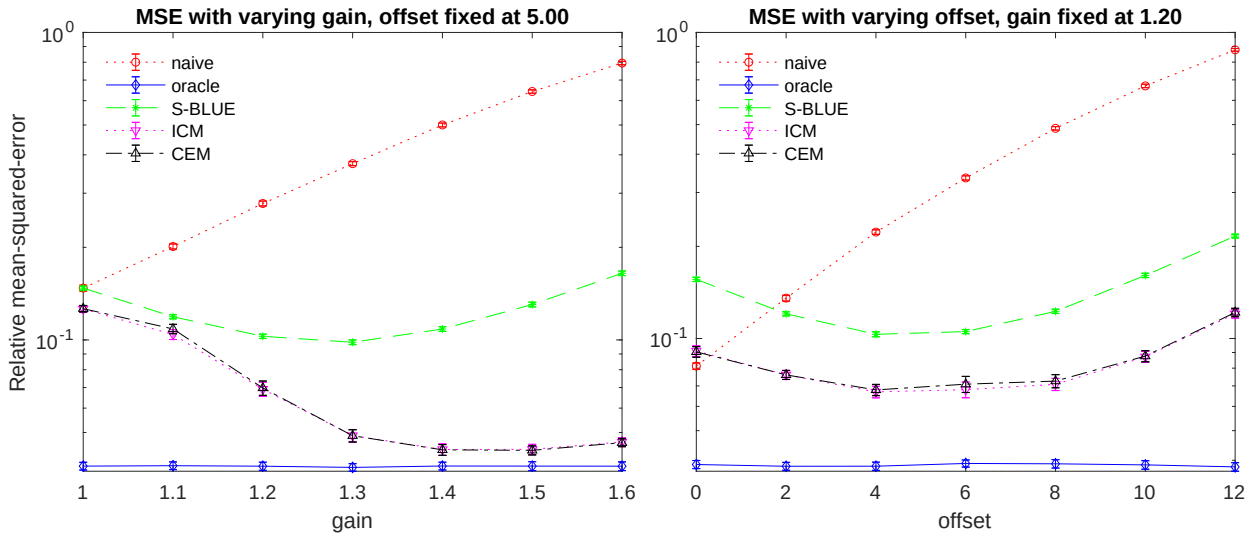


Fig. 3: Synthetic experiment 1 – relative MSE (log-scale) with error bars indicating the 95% confidence interval against varying strengths of distortion.

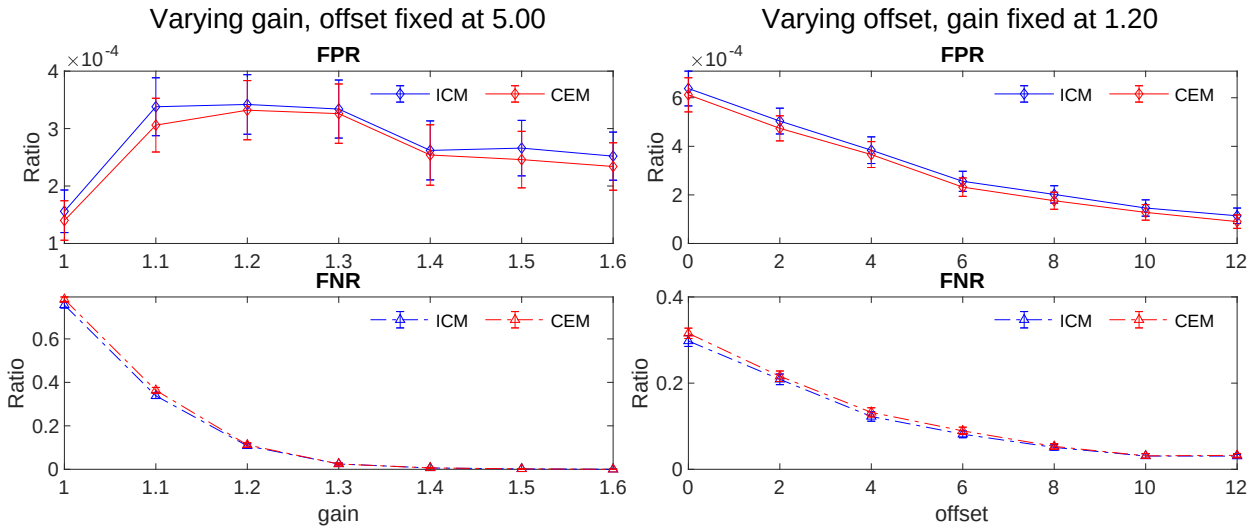


Fig. 4: Synthetic experiment 1 – FPR and FNR of CEM and ICM with error bars indicating the 95% confidence interval against varying strengths of distortion.

randomly selected to have the different distortion parameters generated from the following prior with three categories ($K = 3$), given by $q_1^{(n)} = q_2^{(n)} = q_3^{(n)} = \frac{1}{6}$ for all n, where under π_1, $a_n \sim \log \mathcal{N}(-0.4, 0.05^2)$, $b_n \sim \mathcal{N}(0, 0.2^2)$, under π_2, $a_n \sim \log \mathcal{N}(0.2, 0.05^2)$, $b_n \sim \mathcal{N}(0, 0.2^2)$, under π_3, $a_n \sim \log \mathcal{N}(0, 0.05^2)$, $b_n \sim \mathcal{N}(10, 2^2)$, and the rest of the sensors are set to be non-distorting. The distortion parameters are generated and fixed in this experiment. After that, we randomly simulate 100 sets of noisy observations. For each set of observations, we test the proposed

TABLE I: Synthetic experiment 1 & 2 – maximum absolute deviation from the average relative MSE of the five methods.

Method	Experiment 1	Experiment 2
oracle	0.0278	0.1615
naive	0.1566	0.2064
S-BLUE	0.0410	0.1800
ICM	0.0937	0.1935
CEM	0.0908	0.1928

methods along with the baselines as in the synthetic experiment 1.

Figure 5 shows the relative MSE averaged over 100 realizations, with different number of observations per sensor and different SNR. Figure 6 shows the FPR and FNR of CEM and ICM averaged over 100 realizations. Observe that the MSE of the baselines and S-BLUE did not change with different number of observations per sensor because they depend only on the mean of observations from each sensor. CEM and ICM, on the other hand, benefited from having access to more observations. The naive baseline showed a peculiar trend that first decreased and then increased when SNR increased. The reason is that the naive estimator is a linear combination of the prior mean and the observations, where the weights depend on the SNR. With high SNR, the naive estimator placed a high weight on the distorted observations, thus making the MSE high. The MSE of S-BLUE decreased steadily when SNR increased, and eventually flattened out. The MSE of CEM and ICM depended highly on the number of observations. CEM had the best reconstruction quality among the proposed methods when observations were plentiful or when the SNR was high. In comparison, ICM performed considerably worse than CEM when the number of observations was 5 and 20. This was due to the higher spatial correlation, which made the dependency between distortion parameters higher in the posterior and reduced the effectiveness of iterative greedy search. Notice that the FPR of CEM and ICM was close to 1 when the SNR was low and the observations were scarce. This indicates that CEM and ICM estimated all of the sensors as distorting. The reason was that with high noise variance, the likelihood had a flat shape, and thus the posterior mode did not contain non-distorting sensors. It can also be observed that the FPR and FNR were low when the number of observations was 100 and the SNR was high. Same as in the synthetic experiment 1, the error bars are narrow, indicating the statistical significance of the differences. The third column of Table I shows the maximum absolute deviation from the relative MSE in the synthetic experiment 2. The deviations are

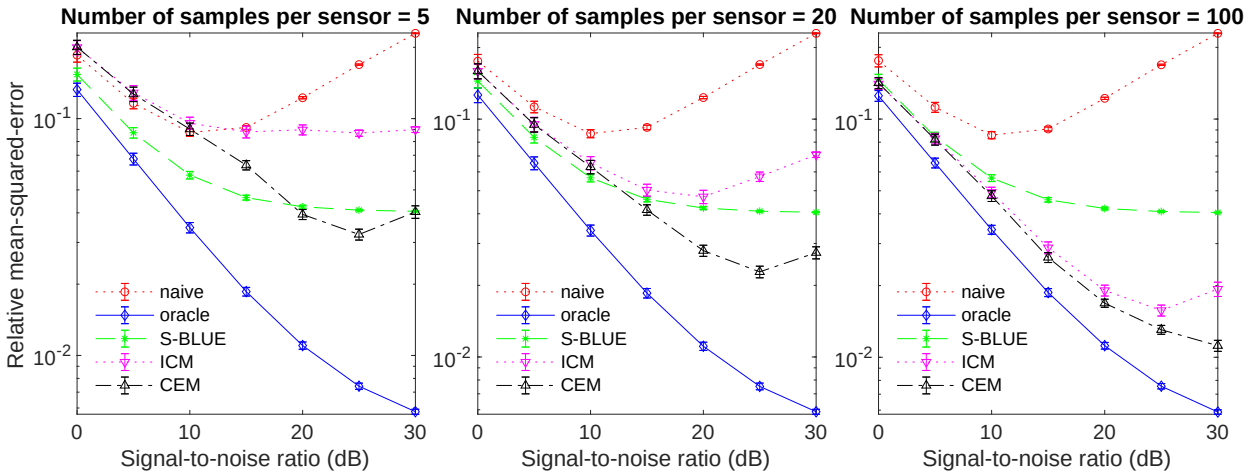


Fig. 5: Synthetic experiment 2 – relative MSE against varying number of observations and SNR.

larger compared to the synthetic experiment 1 due to the case with small number of samples and low SNR. Nonetheless, this shows that the performance of the methods is stable across realizations. To further demonstrate the performance of the proposed methods under different distortion parameters, we repeat the above experiment. This time, instead of fixing the distortion parameters across realizations, the distortion parameters are independently randomly generated in each of the 100 realizations. The result is shown in Figure 7. Since Figure 5 and Figure 7 look very similar, we confirm that the performance is stable across a range of distortion parameters.

Finally, to show the effect of the proportion of distorting sensors, an additional experiment is performed. Figure 8 shows the relative MSE of S-BLUE, CEM, ICM and the baselines when the number of observations is fixed at 100, the SNR is fixed at 20dB, and the proportion of distorting sensors varies from 0 to 1. These results were averaged across 100 independent realizations of distortion parameters and random noises. From Figure 8, it can be observed that when no sensor was distorting, the relative MSE of all methods coincided. However, as the proportion of distorting sensors increased, the relative MSE of the naive baseline increased rapidly, the relative MSE of S-BLUE increased slowly, the relative MSE of CEM and ICM increased only slightly and flattened out eventually, and the oracle, as expected, was unaffected by the change of proportion. The reason is that increasing the proportion of distorting sensors resulted in an increase in prior model uncertainty that negatively affected S-BLUE, while CEM and ICM used the information from the observations to estimate the distortion parameters and thus were only slightly affected.

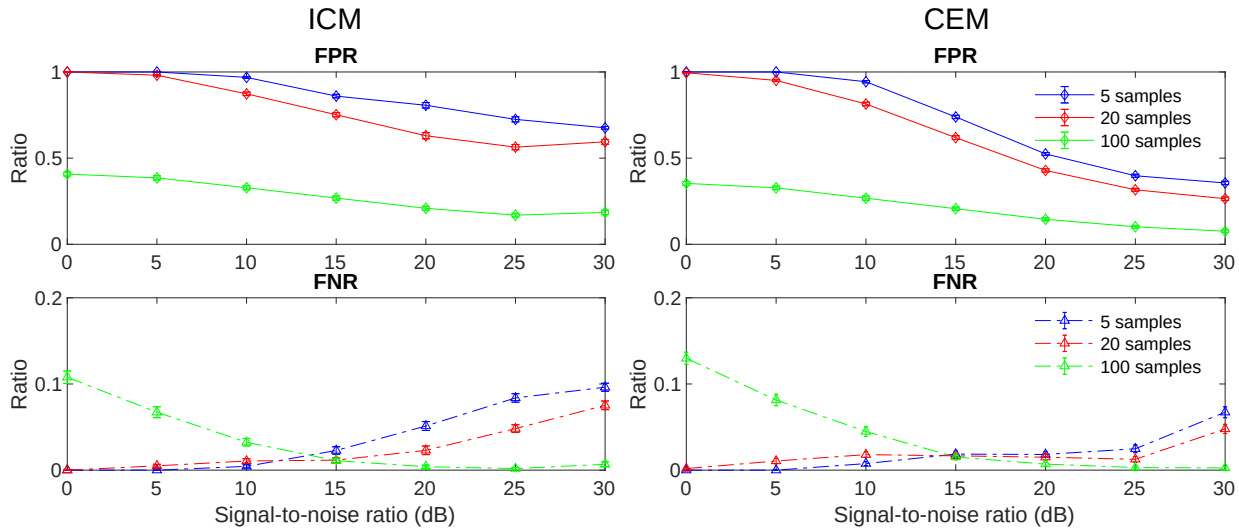


Fig. 6: Synthetic experiment 2 – FPR, FNR of CEM and ICM against varying number of observations and SNR.

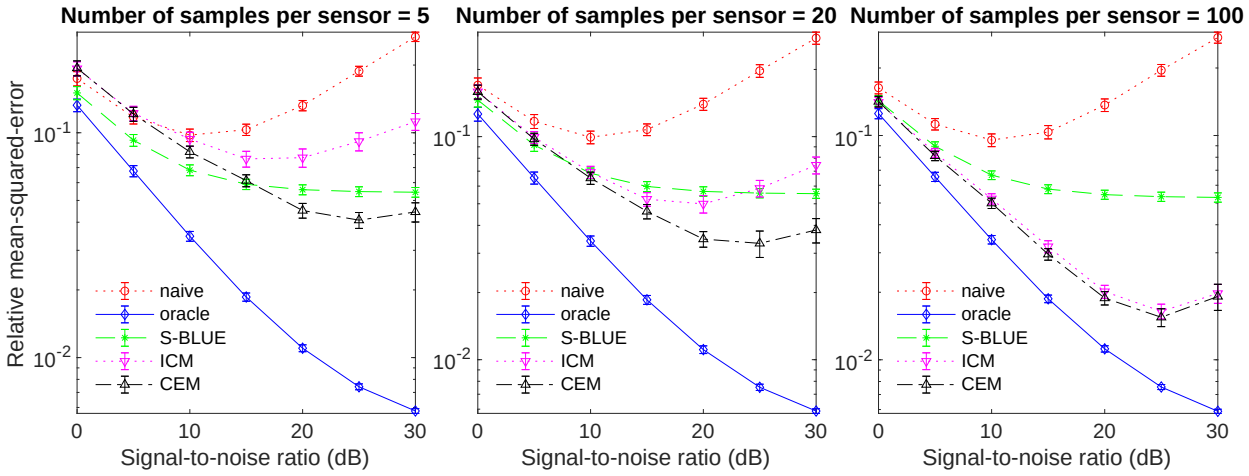


Fig. 7: Synthetic experiment 2 – relative MSE against varying number of observations. Distortion parameters were randomly generated in each of the 100 realizations.

VIII. Experiments with Real Data

We study the 2017 US temperature dataset from the US EPA[2]. The dataset contains 309,226 rows, with the following fields:

- `State.Code`: the numerical code of the state.
- `County.Code`: the numerical code of the county.

[2] https://aqs.epa.gov/aqsweb/airdata/download_files.html, retrieved on 5 June 2018.

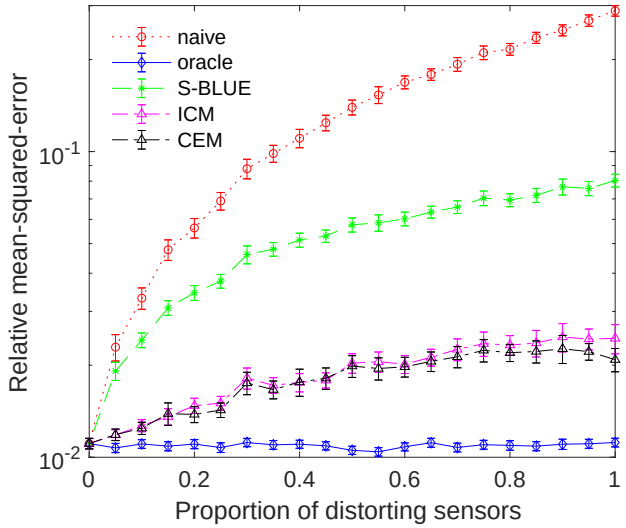


Fig. 8: Synthetic experiment 2 – relative MSE against varying proportion of distorting sensors. The number of observations is fixed at 100 and the SNR is fixed at 20dB.

- `Site.Num`: the numerical code of the monitoring site.
- `Longitude`: the longitude of the site.
- `Latitude`: the latitude of the site.
- `Date.Local`: the local date on which the temperature measurement was taken.
- `X1st.Max.Value`: the maximum hourly temperature measurement of a day.

A. Preprocessing

The first step of preprocessing is to remove the irrelevant fields from the dataset. For temperature measurements, we take the maximum hourly measurement of every day. The measurements are converted from Fahrenheit into Celsius. We noticed that there is an obvious outlier which corresponds to 125°C (the next highest measurement is 55°C) with state code 6, county code 79, site number 5 on day 161. We also noticed another potential outlier corresponding to -17.89°C (0°F) in June, while the next lowest temperature from May to July is 1.1°C. This measurement has state code 38, county code 93, site number 101 on day 181. Therefore, these two measurements are removed from the dataset.

In the dataset, there are 830 monitoring sites in total. Figure 9 is a scatter plot showing the spatial locations of these monitoring sites. Not all monitoring sites have taken measurements on all 365 days. We refer to these as missing measurements. Out of the 830 monitoring sites, 55

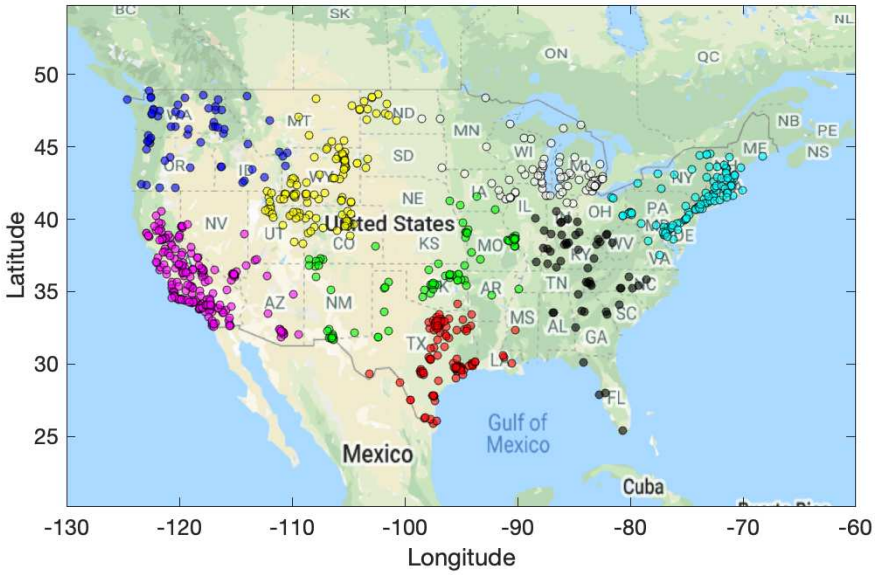
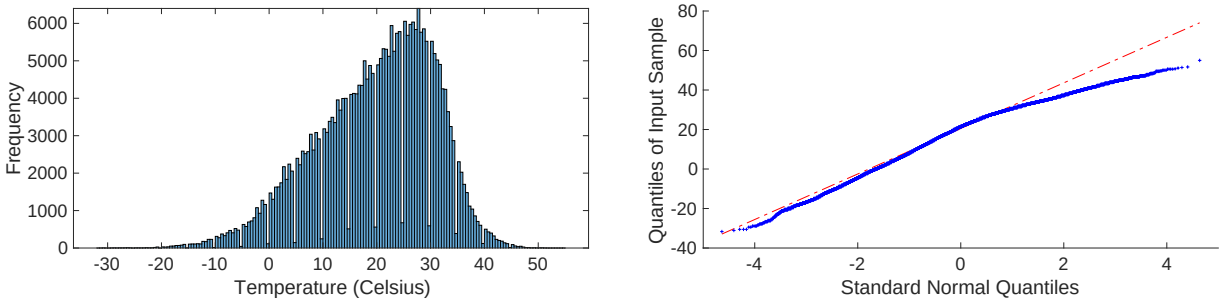


Fig. 9: Spatial locations of temperature monitoring sites. Colors of the points indicate the 8 spatial clusters used to evaluate the distributed version of the proposed methods.



(a) histogram of raw data

(b) normal Q-Q plot

Fig. 10: Histogram and normal Q-Q plot of the temperature measurements.

contain more than 30% of missing measurements. Overall, 16,980 measurements are missing, which is 5.6% of the 302,950 measurement (365×830).

Remark 3. *We have noticed that some monitoring sites have reported temperature measurements by different types of instrument, which explains why we have slightly fewer measurements after we sort measurements into the site & date format. However, measurements on the same day at the same monitoring site with different instruments tend to be close hence we choose an arbitrary set whenever a monitoring site reports multiple sets of measurements.*

Figure 10 shows the histogram and the normal quantile-quantile (Q-Q) plot of the mea-

TABLE II: Summary statistics of the temperature measurements.

mean	variance	skewness	kurtosis	
19.7930	119.9365	-0.5429	2.9377	
min	1st-quartile	median	3rd-quartile	max
-31.6667	12.7778	21.1111	28.3333	55.0000

TABLE III: Estimated GP hyperparameters

signal mean (°C)	signal std. dev. (°C)
30.5034	4.6587
noise std. dev. (°C)	length-scale (km)
1.6340	174.3699

surements. From Figure 10a, one can see that the measurements contain quantization artifacts. Nonetheless, the measurements are treated as continuous. From Figure 10b, one sees that the dataset does not contain abnormally large or small values, and is slightly left-skewed compared to a normal distribution. Overall, the normality assumption holds approximately, as we have not yet considered the seasonal shift of temperature. Table II shows the summary statistics of these measurements.

B. Smoothing of Spatial Field

The first step of the spatial analysis is to model the daily observations as noisy samples from a Gaussian process without distortions and estimate its hyperparameters. For subsequent analyses, we take only data from the 20 days between day 181 to day 200 from the dataset, and restrict ourselves to the contiguous United States (that is, excluding Hawaii and Alaska) since Hawaii and Alaska are far away from the rest of the Unites States. As for the covariance function, we again choose to use the Matérn covariance function with $\nu = 3/2$ to allow for high flexibility while keeping the spatial field mean-square differentiable. For each calendar day, we estimate the signal mean, signal variance, noise variance and length-scale via maximum marginal likelihood estimation (see Chapter 5 of [30] for details) using all available observations on that day. Then, the median of the estimated values on 20 days are taken as the estimated hyperparameters of the GP. The estimated hyperparameters are shown in Table III.

Using the estimated hyperparameters, we reconstruct the spatial field for each calendar day at both a 100×100 grid of locations and the 824 sensor locations by computing the posterior

mean and the posterior covariance matrix, conditional on all the available observations on that day. Since the posterior mean is usually much smoother compared to the actual spatial field, a sample spatial field is generated from the posterior distribution to make it realistic. The generated spatial field at a grid of locations is treated as the ground truth of the spatial field, and the generated field intensities at the 824 sensor locations are treated as as the noise-free sensor reading from which noisy and distorted observations are generated.

C. Experimental Settings

In this experiment, in addition to the five methods tested in the synthetic experiments, we evaluate the performance of the distributed version of S-BLUE, CEM, and ICM introduced in Section VI. To do so, we first divide the 824 sensor locations into 8 disjoint clusters via hierarchical clustering with great-circle distance and complete linkage (see e.g. Section 14.3.12 of [35]). The 8 resulting clusters are indicated in Figure 9 using 8 different colors. Subsequently, we apply the distributed approaches to this partitioned sensor network.

In order to evaluate the reconstruction accuracy of the proposed methods, 25% of the sensor locations are left out as the test set. For the 618 remaining locations, we randomly generate the distortion parameters from three different settings. In the three settings, each sensor has a respective probability of 0.3, 0.5 and 0.7 to introduce distortion. Under all three settings, the distorting sensors have one of the three following categories, with equal probabilities. Under π_1, $a \sim \log\mathcal{N}(-0.4, 0.05^2)$, $b \sim \mathcal{N}(0, 0.2^2)$, under π_2, $a \sim \log\mathcal{N}(0.2, 0.05^2)$, $b \sim \mathcal{N}(0, 0.2^2)$, under π_3, $a \sim \log\mathcal{N}(0, 0.05^2)$, $b \sim \mathcal{N}(10, 2^2)$. For each of the three settings, we first randomly generate the distortion parameters for each sensor. Then we randomly simulate noisy observations at each sensor location and subsequently apply the corresponding distortions. In addition, we examine the effect of the number of observations and effective SNR. Specifically, we examine the four following cases:

1) 10 observations per sensor, SNR=5dB;
2) 10 observations per sensor, SNR=15dB;
3) 50 observations per sensor, SNR=5dB;
4) 50 observations per sensor, SNR=15dB.

D. Results and Discussion

Figure 11 shows the relative MSE averaged over 20 days, and Figure 12 shows the FPR and FNR of CEM and ICM, under each setting. Similar to the synthetic experiment 2, we observe that

the MSE of the baselines and S-BLUE did not change with different number of observations per sensor. CEM and ICM, on the other hand, benefited from having access to more observations.

The naive baseline showed worse MSE when SNR was high. The reason was the same as in the synthetic experiment 2. In addition, the MSE of naive baseline is clearly affected by a higher proportion of distorting sensors.

S-BLUE showed stable accuracies across all settings. Since S-BLUE does not estimate the distortion parameters, the error mainly resulted from smoothing. This can be seen clearly from the reconstructed spatial fields in Figure 13, which will be discussed later.

Compared to the simple method S-BLUE, the more sophisticated methods CEM and ICM have the additional benefit of being able to estimate the distortion parameters. CEM consistently outperformed ICM, which is consistent with what we observed in Section VII-B. This possibly indicates the ineffectiveness of the iterative greedy search method with high-dimensional mixed discrete-continuous optimization problems. While CEM benefited slightly from higher SNR, the performance of ICM deteriorated with higher SNR when there were 10 observations per sensor, though they both benefited from more plentiful observations. Looking at Figure 12, it can be observed that ICM had significantly higher FPR and slightly lower FNR compared to CEM. More observations and higher SNR have the effect of decreasing the FPR of both CEM and ICM.

Remark 4. *We noticed that if the SNR is set to be higher than 20dB, the performance of both CEM and ICM deteriorates greatly. With a high SNR (hence low noise variance), the posterior density of the model tends to be highly irregular. Thus the optimization problem might be highly ill-posed. In general, there is no universal solution to this. Some form of relaxation might help to improve the performance, but that is not examined in this work.*

In the low SNR settings, the distributed approaches performed slightly worse compared to their centralized counterparts. In the high SNR settings, the distributed S-BLUE performed slightly worse compared to the centralized S-BLUE. However, the distributed CEM and ICM performed better compared to their centralized version. This indicates that both CEM and ICM suffered from convergence issues when dealing with the high-dimensional optimization problem. In the distributed version, the problem is decomposed into sub-problems of lower dimensionality, which are easier to solve. This experiment shows that the distributed approaches we proposed are the ideal candidates for solving such spatial field reconstruction problem in large-scale settings.

Table IV shows the maximum absolute deviation from the relative MSE in this experiment. It shows that S-BLUE and the distributed CEM had stable performance throughout the 20 days. ICM and the centralized CEM are less stable, presumably due to the difficulty in the optimization procedures.

TABLE IV: Experiment with real data – maximum absolute deviation from the average relative MSE of the eight methods.

Method	Centralized	Distributed
oracle	0.0728	-
naive	0.2015	-
S-BLUE	0.1132	0.1128
ICM	0.2676	0.3188
CEM	0.1838	0.1104

Finally, let us examine the reconstructed spatial fields. Figure 13 shows the heat maps of spatial fields reconstructed by centralized and distributed versions of S-BLUE, CEM, ICM, and the two baselines with the settings: proportion of distorting sensors is 0.7, 10 observations per sensor, SNR=15dB. The ground truth spatial field is also included for reference. First, notice that the ground truth contained much more details compared to the reconstructions, because all of the estimators here have the smoothing effect. Overall, the reconstruction of S-BLUE is much smoother and contains fewer details. This is due to nature of S-BLUE, as it does not estimate the distortion parameters. CEM and ICM, on the other hand, preserved many of the details, and their reconstructions were overall close to the one produced by the oracle. The naive method, however, produced a noticeably inaccurate reconstruction. As previously mentioned, the reconstructions of the distributed approaches are discontinuous and the discontinuity can be observed at the boundary of the clusters. Nonetheless, they produced accurate reconstructions of the spatial field.

Notice that the above analysis used only 20 consecutive days of data. When a longer time period is considered, the seasonal variation of the underlying spatial field must be taken into consideration. To demonstrate this, we use the data from each of the twelve months to estimate the GP hyperparameters by the same approach as above. Subsequently, the spatial field on the last day of each month is reconstructed by centralized CEM, using noisy and distorted sensor readings at the 618 locations generated in the same way as in the above experiment. These reconstructions along with the ground truths are shown in Figure 14. It can be seen

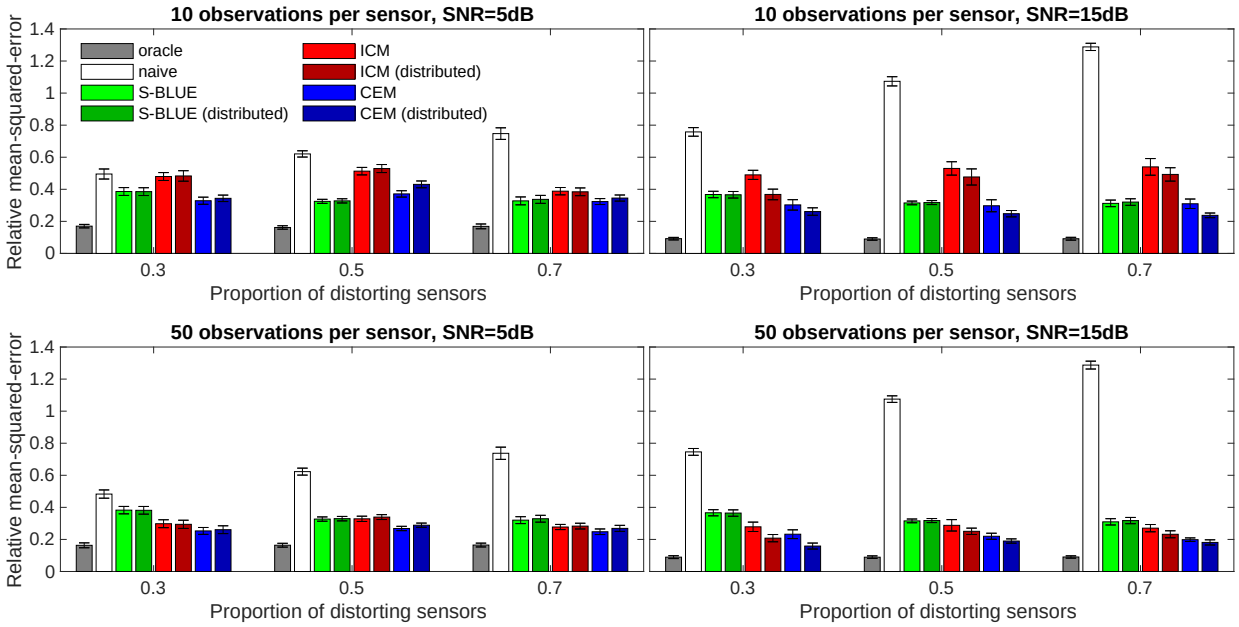


Fig. 11: Experiment with real data – relative MSE against varying proportion of distorting sensors, number of observations and SNR.

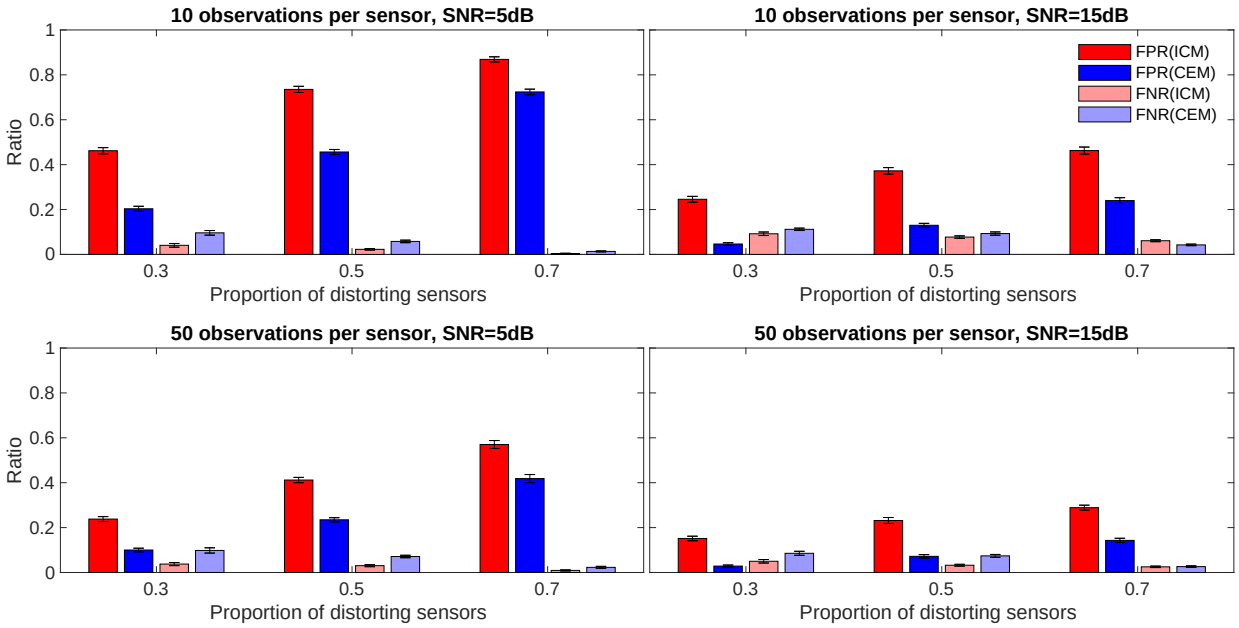


Fig. 12: Experiment with real data – FPR, FNR of CEM and ICM.

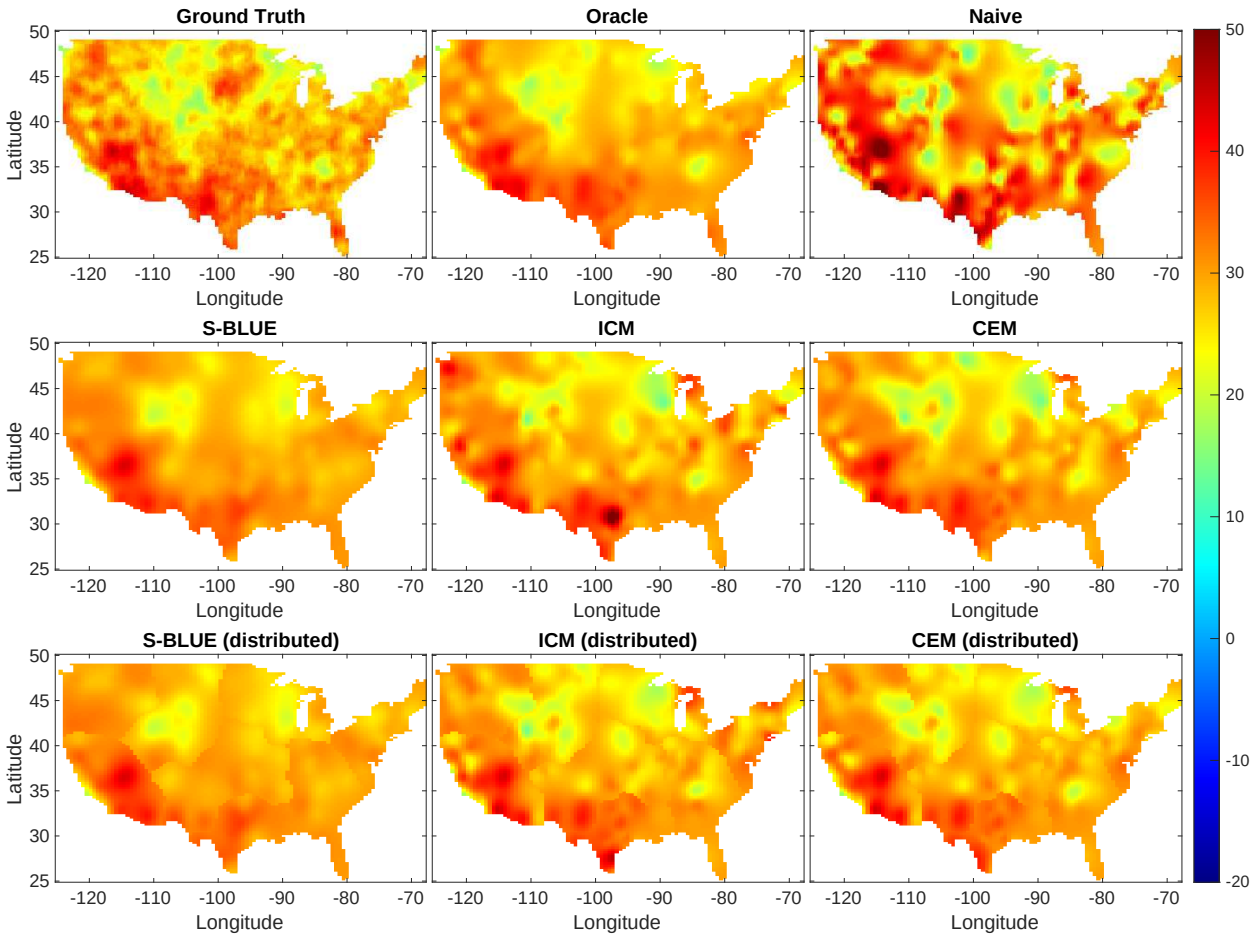


Fig. 13: Experiment with real data – Reconstructed spatial fields on day 181. Settings: proportion of distorting sensors is 0.7, 10 observations per sensor, SNR=15dB.

that there is an obvious seasonal effect on the temperature throughout the year. Notice that the reconstructions closely resemble the ground truths, which indicates that CEM worked well when the characteristics of the underlying spatial field varied throughout the year.

IX. Conclusion

This paper addressed the problem of spatial field reconstruction based on distorted sensor readings. A new spatial field model based on a mixture of Gaussian process experts was developed. We developed two approaches to solve the inference problem. The first approach uses a linear Bayes estimator named the Spatial Best Linear Unbiased Estimator (S-BLUE), which is a low-complexity algorithm relying only on prior information. The second approach is a two-stage algorithm based on empirical Bayes, in which the unknown distortion parameters of the sensors are estimated based on distorted observations. We developed two optimization procedures for

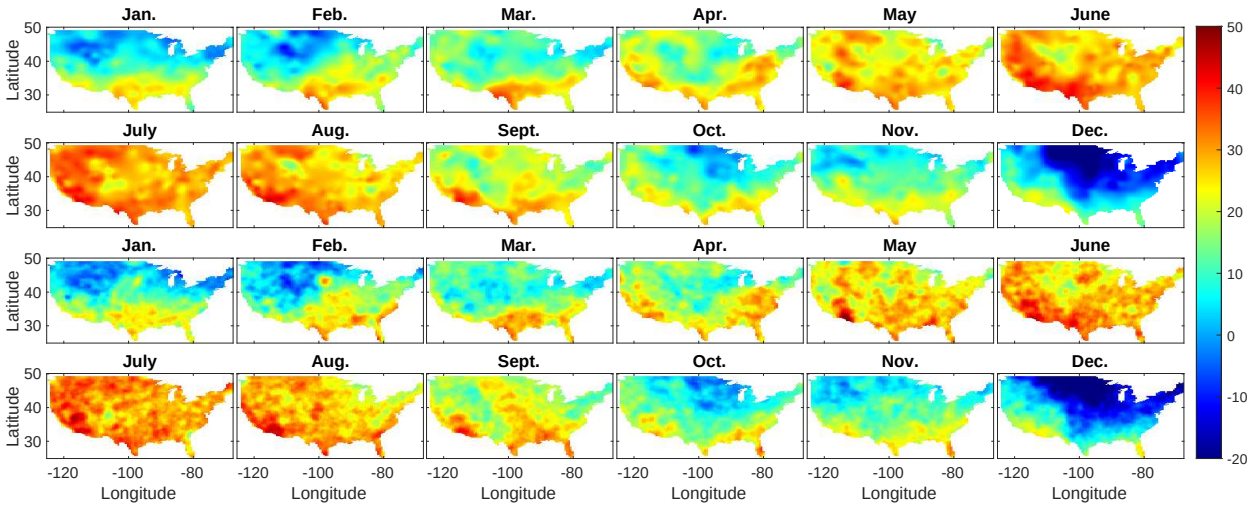


Fig. 14: Experiment with real data – Reconstructed spatial fields on the last day of each month. The two top rows show the reconstructions by CEM, and the two bottom rows show the ground truths. Settings: proportion of distorting sensors is 0.7, 10 observations per sensor, SNR=15dB.

the two-stage algorithm, the first one is based on the Cross-Entropy method (CEM) and the second one is based on the Iterated Conditional Mode (ICM) which is an iterative greedy search procedure. In addition, the distributed versions of S-BLUE and empirical Bayes estimators were developed to improve the computational efficiency in large-scale applications. We preformed two synthetic experiments as well as an experiment based on real temperature data from US EPA with synthetically generated distortions to assess the spatial field reconstruction accuracy of the proposed approaches. The results showed significant improvement compared to the estimation approach that neglects sensor distortions.

ACKNOWLEDGMENTS

The research was conducted under the Cooling Singapore project, funded by Singapores National Research Foundation (NRF) under its Virtual Singapore programme.

REFERENCES

[1] I. Nevat, G. W. Peters, F. Septier, and T. Matsui, "Estimation of Spatially Correlated Random Fields in Heterogeneous Wireless Sensor Networks," *IEEE Transactions on Signal Processing*, vol. 63, no. 10, pp. 2597–2609, 2015. 2, 3, 4

[2] C. Sun, Y. Yu, V. O. Li, and J. C. Lam, "Optimal multi-type sensor placements in gaussian spatial fields for environmental monitoring," in *2018 IEEE International Smart Cities Conference (ISC2)*. IEEE, 2018, pp. 1–8. 2, 3, 4

[3] H. Sheng, J. Xiao, Y. Cheng, Q. Ni, and S. Wang, "Short-term solar power forecasting based on weighted gaussian process regression," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 1, pp. 300–308, 2018. 2, 4

[4] K. Sohraby, D. Minoli, and T. Znati, *Wireless sensor networks: technology, protocols, and applications*. John Wiley & Sons, 2007. 2

[5] E. Soltanmohammadi, M. Orooji, and M. Naraghi-Pour, "Decentralized hypothesis testing in wireless sensor networks in the presence of misbehaving nodes," *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 1, pp. 205–215, 2013. 2

[6] B. Ristic, D. E. Clark, and N. Gordon, "Calibration of multi-target tracking algorithms using non-cooperative targets," *IEEE Journal of Selected Topics in Signal Processing*, vol. 7, no. 3, pp. 390–398, 2013. 2

[7] T. Watkins, "Draft roadmap for next generation air monitoring," *Environmental Protection Agency*, 2013. 2

[8] A. Arfire, A. Marjovi, and A. Martinoli, "Model-based rendezvous calibration of mobile sensor networks for monitoring air quality," in *2015 IEEE SENSORS*. IEEE, 2015, pp. 1–4. 2

[9] F. Fazel, M. Fazel, and M. Stojanovic, "Random access sensor networks: Field reconstruction from incomplete data," in *2012 Information Theory and Applications Workshop*. IEEE, 2012, pp. 300–305. 2

[10] F. Restuccia, N. Ghosh, S. Bhattacharjee, S. K. Das, and T. Melodia, "Quality of information in mobile crowdsensing: Survey and research challenges," *ACM Transactions on Sensor Networks (TOSN)*, vol. 13, no. 4, p. 34, 2017. 2

[11] X. Fang and I. Bate, "Using multi-parameters for calibration of low-cost sensors in urban environment," in *Proceedings of the 2017 International Conference on Embedded Wireless Systems and Networks*. Junction Publishing, 2017, pp. 1–11. 2

[12] T. R. Karl, A. Arguez, B. Huang, J. H. Lawrimore, J. R. McMahon, M. J. Menne, T. C. Peterson, R. S. Vose, and H.-M. Zhang, "Possible artifacts of data biases in the recent global surface warming hiatus," *Science*, vol. 348, no. 6242, pp. 1469–1472, 2015. 2

[13] Ç. Bilen, G. Puy, R. Gribonval, and L. Daudet, "Convex optimization approaches for blind sensor calibration using sparsity," *IEEE Transactions on Signal Processing*, vol. 62, no. 18, pp. 4847–4856, 2014. 2

[14] O. Saukh, D. Hasenfratz, C. Walser, and L. Thiele, "On rendezvous in mobile sensing networks," in *Real-World Wireless Sensor Networks*. Springer, 2014, pp. 29–42. 2, 6

[15] C. Dorffer, M. Puigt, G. Delmaire, and G. Roussel, "Informed nonnegative matrix factorization methods for mobile sensor network calibration," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 4, no. 4, pp. 667–682, 2018. 2, 6

[16] Y. C. Eldar, W. Liao, and S. Tang, "Sensor calibration for off-the-grid spectral estimation," *Applied and Computational Harmonic Analysis*, 2018. 2

[17] M. Cho, W. Liao, and Y. Chi, "A non-convex approach to joint sensor calibration and spectrum estimation," in *2018 IEEE Statistical Signal Processing Workshop (SSP)*. IEEE, 2018, pp. 398–402. 2

[18] E. Nurellari, D. McLernon, and M. Ghogho, "A secure optimum distributed detection scheme in under-attack wireless sensor networks," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 4, no. 2, pp. 325–337, 2018. 2

[19] A. S. Rawat, P. Anand, H. Chen, and P. K. Varshney, "Collaborative spectrum sensing in the presence of byzantine attacks in cognitive radio networks," *IEEE Transactions on Signal Processing*, vol. 59, no. 2, pp. 774–786, 2010. 3

[20] P. Zhang, J. Y. Koh, S. Lin, and I. Nevat, "Distributed event detection under byzantine attack in wireless sensor

networks," in *2014 IEEE Ninth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP)*. IEEE, 2014, pp. 1–6. 3

[21] T. Monahan and J. T. Mokos, "Crowdsourcing urban surveillance: The development of homeland security markets for environmental sensor networks," *Geoforum*, vol. 49, pp. 279–288, 2013. 3

[22] E. Arias-de Reyna, P. Closas, D. Dardari, and P. M. Djuric, "Crowd-based learning of spatial fields for the internet of things: From harvesting of data to inference," *IEEE Signal Processing Magazine*, vol. 35, no. 5, pp. 130–139, 2018. 3

[23] P. Zhang, I. Nevat, G. W. Peters, F. Septier, and M. A. Osborne, "Spatial field reconstruction and sensor selection in heterogeneous sensor networks with stochastic energy harvesting," *IEEE Transactions on Signal Processing*, vol. 66, no. 9, pp. 2245–2257, 2018. 3

[24] G. W. Peters, I. Nevat, and T. Matsui, "How to Utilize Sensor Network Data to Efficiently Perform Model Calibration and Spatial Field Reconstruction," in *Modern Methodology and Applications in Spatial-Temporal Modeling*. Springer, 2015, pp. 25–62. 3

[25] I. Nevat, G. W. Peters, and I. B. Collings, "Random Field Reconstruction With Quantization in Wireless Sensor Networks," *IEEE Transactions on Signal Processing*, vol. 61, pp. 6020–6033, 2013. 3, 4

[26] J. Unnikrishnan and M. Vetterli, "Sampling and reconstruction of spatial fields using mobile sensors," *IEEE Transactions on Signal Processing*, vol. 61, no. 9, pp. 2328–2340, 2013. 3

[27] I. Koukoutsidis, "Estimating spatial averages of environmental parameters based on mobile crowdsensing," *ACM Transactions on Sensor Networks (TOSN)*, vol. 14, no. 1, pp. 1–26, 2017. 3

[28] Q. Xiang, J. Zhang, I. Nevat, and P. Zhang, "A trust-based mixture of gaussian processes model for reliable regression in participatory sensing." in *IJCAI*, 2017, pp. 3866–3872. 3, 4

[29] I. Nevat, G. W. Peters, and I. B. Collings, "Location-aware cooperative spectrum sensing via Gaussian Processes," in *Communications Theory Workshop (AusCTW), 2012 Australian*. IEEE, 2012, pp. 19–24. 4

[30] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2005. 5, 11, 32

[31] E. Miluzzo, N. D. Lane, A. T. Campbell, and R. Olfati-Saber, "Calibree: A self-calibration system for mobile sensor networks," in *International Conference on Distributed Computing in Sensor Systems*. Springer, 2008, pp. 314–331. 6

[32] R. Rubinstein, "The cross-entropy method for combinatorial and continuous optimization," *Methodology and computing in applied probability*, vol. 1, no. 2, pp. 127–190, 1999. 14

[33] J. A. Bilmes *et al.*, "A gentle tutorial of the em algorithm and its application to parameter estimation for gaussian mixture and hidden markov models," 1998. 16

[34] J. Besag, "On the statistical analysis of dirty pictures," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 48, no. 3, pp. 259–279, 1986. 17

[35] J. Friedman, T. Hastie, and R. Tibshirani, *The elements of statistical learning*. Springer series in statistics New York, 2001, vol. 1, no. 10. 33

APPENDIX A
PROOF OF THEOREMS

A. Proof of Theorem 1

Proof. For $n = 1, \ldots, N$, the log model likelihood given $\boldsymbol{f} := (f(\mathbf{x}_n))_{n=1:N}$ is given by

$$\begin{aligned}
&\log p(y_{n,1:M_n}|\boldsymbol{f}, \boldsymbol{\psi}) \\
=& -\frac{M_n}{2}\log 2\pi - \frac{M_n}{2}\log a_n^2\varsigma^2 - \frac{s_n}{2a_n^2\varsigma^2} \\
&+ \frac{(a_n f_n + b_n)g_n}{a_n^2\varsigma^2} - \frac{M_n(a_n f_n + b_n)^2}{2a_n^2\varsigma^2},
\end{aligned}$$

which depends on $y_{n,1:M_n}$ only through the statistics g_n and s_n. Therefore, $(\boldsymbol{g}, \boldsymbol{s})$ are sufficient for $(\boldsymbol{f}, \boldsymbol{\psi})$. The joint density of $(\boldsymbol{y}, \boldsymbol{f})$ conditional on $\boldsymbol{\psi}$ is given by

$$\begin{aligned}
&\log p(\boldsymbol{y}, \boldsymbol{f}|\boldsymbol{\psi}) \\
=& \log p(\boldsymbol{y}|\boldsymbol{f}) + \log p(\boldsymbol{f}|\boldsymbol{\psi}) \\
=& -\frac{1}{2}\Big[(N + \mathrm{tr}(\mathbf{M}))\log 2\pi + \mathrm{tr}(\mathbf{M}\log(\varsigma^2\mathbf{A}^2)) + \log|\boldsymbol{\mathcal{C}}| \\
&+ \varsigma^{-2}\mathbf{1}^T\mathbf{A}^{-2}\boldsymbol{s} + \boldsymbol{\mu}^T\boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mu} + \varsigma^{-2}\boldsymbol{b}^T\mathbf{M}\mathbf{A}^{-2}\boldsymbol{b} - \boldsymbol{\gamma}^T\mathbf{Z}^{-1}\boldsymbol{\gamma} \\
&- 2\varsigma^{-2}\boldsymbol{g}^T\mathbf{A}^{-2}\boldsymbol{b} + \left(\boldsymbol{f} - \mathbf{Z}^{-1}\boldsymbol{\gamma}\right)^T\mathbf{Z}\left(\boldsymbol{f} - \mathbf{Z}^{-1}\boldsymbol{\gamma}\right)\Big],
\end{aligned} \tag{30}$$

where $\boldsymbol{\gamma} = \varsigma^{-2}\mathbf{M}\tilde{\boldsymbol{g}} + \boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mu}, \mathbf{Z} = \varsigma^{-2}\mathbf{M} + \boldsymbol{\mathcal{C}}^{-1}$. Integrating over $\boldsymbol{f}$, we deduce that,

$$\begin{aligned}
\log p(\boldsymbol{y}|\boldsymbol{\psi}) =& \log\left[\int p(\boldsymbol{y}, \boldsymbol{f}|\boldsymbol{\psi})d\boldsymbol{f}\right] \\
=& -\frac{1}{2}\left[\mathrm{tr}(\mathbf{M})\log 2\pi + \mathrm{tr}(\mathbf{M}\log(\varsigma^2\mathbf{A}^2))\right. \\
&+ \log|\boldsymbol{\mathcal{C}}| + \log|\mathbf{Z}| + \varsigma^{-2}\mathbf{1}^T\mathbf{A}^{-2}\boldsymbol{s} + \boldsymbol{\mu}^T\boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mu} \\
&\left.+\varsigma^{-2}\boldsymbol{b}^T\mathbf{M}\mathbf{A}^{-2}\boldsymbol{b} - 2\varsigma^{-2}\boldsymbol{g}^T\mathbf{A}^{-2}\boldsymbol{b} - \boldsymbol{\gamma}^T\mathbf{Z}^{-1}\boldsymbol{\gamma}\right].
\end{aligned} \tag{31}$$

We have by Woodbury's formula that,

$$\mathbf{Z}^{-1} = \varsigma^2\mathbf{M}^{-1} - \varsigma^4\mathbf{M}^{-1}(\boldsymbol{\mathcal{C}} + \varsigma^2\mathbf{M}^{-1})^{-1}\mathbf{M}^{-1} \tag{32}$$

$$= \boldsymbol{\mathcal{C}} - \boldsymbol{\mathcal{C}}(\boldsymbol{\mathcal{C}} + \varsigma^2\mathbf{M}^{-1})^{-1}\boldsymbol{\mathcal{C}}, \tag{33}$$

$$\log|\mathbf{Z}| = \log|\varsigma^{-2}\mathbf{M}| - \log|\boldsymbol{\mathcal{C}}| + \log|\boldsymbol{\mathcal{C}} + \varsigma^2\mathbf{M}^{-1}|. \tag{34}$$

We also have by $\boldsymbol{\gamma} = \varsigma^{-2}\mathbf{M}\tilde{\boldsymbol{g}} + \boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mu}$ that,

$$\begin{aligned}
\boldsymbol{\gamma}^T\mathbf{Z}^{-1}\boldsymbol{\gamma} =& \varsigma^{-4}\tilde{\boldsymbol{g}}^T\mathbf{M}\mathbf{Z}^{-1}\mathbf{M}\tilde{\boldsymbol{g}} + 2\varsigma^{-2}\tilde{\boldsymbol{g}}^T\mathbf{M}\mathbf{Z}^{-1}\boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mu} \\
&+ \boldsymbol{\mu}^T\boldsymbol{\mathcal{C}}^{-1}\mathbf{Z}^{-1}\boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mu}.
\end{aligned} \tag{35}$$

Substituting (32) into the first term on the right-hand side of (35), using the fact that $\mathbf{M}\mathbf{Z}^{-1}\boldsymbol{\mathcal{C}}^{-1} = (\boldsymbol{\mathcal{C}}\mathbf{Z}\mathbf{M}^{-1})^{-1} = (\varsigma^{-2}\boldsymbol{\mathcal{C}} + \mathbf{M}^{-1})^{-1} = \varsigma^2\boldsymbol{\Upsilon}^{-1}$ for the second term of (35), substituting (33) into the third term of (35), and finally substituting the resulting formula as well as (34) into (31), one gets the following after simplification (recall that $\tilde{\boldsymbol{g}} := \mathbf{A}^{-1}(\mathbf{M}^{-1}\boldsymbol{g} - \boldsymbol{b})$, $\boldsymbol{\Upsilon} := \boldsymbol{\mathcal{C}} + \varsigma^2\mathbf{M}^{-1}$),

$$\begin{aligned}\log p(\boldsymbol{y}|\boldsymbol{\psi}) = -\frac{1}{2}\big[&\mathrm{tr}(\mathbf{M})\log 2\pi + \mathrm{tr}(\mathbf{M}\log(\varsigma^2\mathbf{A}^2)) \\ &- \log|\varsigma^2\mathbf{M}^{-1}| + \log|\boldsymbol{\Upsilon}| + \varsigma^{-2}\mathbf{1}^T\mathbf{A}^{-2}\boldsymbol{s} \\ &-\varsigma^{-2}\boldsymbol{g}^T\mathbf{M}^{-1}\mathbf{A}^{-2}\boldsymbol{g} + (\tilde{\boldsymbol{g}} - \boldsymbol{\mu})^T\boldsymbol{\Upsilon}^{-1}(\tilde{\boldsymbol{g}} - \boldsymbol{\mu})\big]\,.\end{aligned}$$

Notice that after integrating out $\boldsymbol{f}$, $(\boldsymbol{g}, \boldsymbol{s})$ are still sufficient for $\boldsymbol{\psi}$.

By Bayes' rule, $p(\boldsymbol{\psi}|\boldsymbol{y}) = \frac{p(\boldsymbol{y}|\boldsymbol{\psi})\pi(\boldsymbol{\psi})}{p(\boldsymbol{y})}$, where $p(\boldsymbol{y}) = \int p(\boldsymbol{y}|\boldsymbol{\psi})\pi(\boldsymbol{\psi})d\boldsymbol{\psi}$ is the normalizing constant that is analytically intractable. The proof is now complete. □

B. Proof of Theorem 2

From (30) we see that $\log p(\boldsymbol{y}, \boldsymbol{f}|\boldsymbol{\psi})$ has the following form,

$$\log p(\boldsymbol{y}, \boldsymbol{f}|\boldsymbol{\psi}) = q_1(\boldsymbol{\psi}) + q_2(\boldsymbol{s}, \boldsymbol{\psi}) + q_3(\boldsymbol{g}, \boldsymbol{f}, \boldsymbol{\psi}),$$

where q_1, q_2, q_3 are some functions of the corresponding parameters. Similarly, we can deduce that $\log p(\boldsymbol{y}, \boldsymbol{f}, f_*|\boldsymbol{\psi})$ has the following form,

$$\log p(\boldsymbol{y}, \boldsymbol{f}, f_*|\boldsymbol{\psi}) = q_1(\boldsymbol{\psi}) + q_2(\boldsymbol{s}, \boldsymbol{\psi}) + q_3(\boldsymbol{g}, [\boldsymbol{f}, f_*], \boldsymbol{\psi}).$$

Therefore,

$$\begin{aligned}p(f_*|\boldsymbol{y}, \boldsymbol{\psi}) =& \frac{p(\boldsymbol{y}, f_*|\boldsymbol{\psi})}{p(\boldsymbol{y}|\boldsymbol{\psi})} = \frac{\int p(\boldsymbol{y}, \boldsymbol{f}, f_*|\boldsymbol{\psi})d\boldsymbol{f}}{\int p(\boldsymbol{y}, \boldsymbol{f}|\boldsymbol{\psi})d\boldsymbol{f}} \\ =& \frac{\int \exp(q_3(\boldsymbol{g}, [\boldsymbol{f}, f_*], \boldsymbol{\psi}))d\boldsymbol{f}}{\int \exp(q_3(\boldsymbol{g}, \boldsymbol{f}, \boldsymbol{\psi}))d\boldsymbol{f}}.\end{aligned} \tag{36}$$

Thus, we have deduced that $p(f_*|\boldsymbol{y}, \boldsymbol{\psi})$ depends on $\boldsymbol{y}$ only through $\boldsymbol{g}$ (or equivalently, $\tilde{\boldsymbol{g}}$), i.e. $p(f_*|\boldsymbol{y}, \boldsymbol{\psi}) = p(f_*|\boldsymbol{g}, \boldsymbol{\psi}) = p(f_*|\tilde{\boldsymbol{g}}, \boldsymbol{\psi})$. We also have that conditional on $\boldsymbol{\psi}$, $(\tilde{\boldsymbol{g}}, f_*)$ are jointly Gaussian, with $\mathbb{E}[f_*|\boldsymbol{\psi}] = \mu_*$, $\mathbb{E}[\tilde{\boldsymbol{g}}|\boldsymbol{\psi}] = \boldsymbol{\mu}$, $\mathrm{Cov}[f_*|\boldsymbol{\psi}] = \mathcal{C}_*$, $\mathrm{Cov}[\tilde{\boldsymbol{g}}, f_*|\boldsymbol{\psi}] = \boldsymbol{k}_*$, $\mathrm{Cov}[\tilde{\boldsymbol{g}}|\boldsymbol{\psi}] = \boldsymbol{\Upsilon}$, which can be easily verified. Thus, we have that $(f_*|\boldsymbol{y}, \boldsymbol{\psi}) \sim \mathcal{N}(\bar{f}_*, \sigma_*^2)$, where $\bar{f}_* = \mu_* + \boldsymbol{k}_*^T\boldsymbol{\Upsilon}^{-1}(\tilde{\boldsymbol{g}} - \boldsymbol{\mu})$, $\sigma_*^2 = \mathcal{C}_* - \boldsymbol{k}_*^T\boldsymbol{\Upsilon}^{-1}\boldsymbol{k}_*$. The distribution of $(f_*, \boldsymbol{\psi}|\boldsymbol{y})$ is given by $p(f_*, \boldsymbol{\psi}|\boldsymbol{y}) = p(f_*|\boldsymbol{y}, \boldsymbol{\psi})p(\boldsymbol{\psi}|\boldsymbol{y})$, and the posterior predictive distribution $(f_*|\boldsymbol{y})$ is obtained by marginalizing over $\boldsymbol{\psi}$, $p(f_*|\boldsymbol{y}) = \int p(f_*|\boldsymbol{y}, \boldsymbol{\psi})p(\boldsymbol{\psi}|\boldsymbol{y})d\boldsymbol{\psi}$. The proof is now complete.

C. Proof of Theorem 3

Proof. First, we claim that any linear estimator must have the form $h(\boldsymbol{y}) = \boldsymbol{w}^T\bar{\boldsymbol{g}} + b$, where $\boldsymbol{w} \in \mathbb{R}^N, b \in \mathbb{R}$. Due to the linearity of h in observations $(y_{n,m})_{n=1:N,m=1:M_n}$, $\boldsymbol{s}$ is not involved. For $n = 1, \ldots, N$, the weights of $(y_{n,m})_{m=1:M_n}$ must be the same due to symmetry. This proves the claim.

Under quadratic loss, for $h \in \mathcal{H}$, $R[\Pi, h]$ is given by

$$\begin{aligned} R[\Pi, h] =& \mathbb{E}\left[\left(\boldsymbol{w}^T\bar{\boldsymbol{g}} + b - f_*\right)^2\right] \\ =& \boldsymbol{w}^T\mathbb{E}[\bar{\boldsymbol{g}}\bar{\boldsymbol{g}}^T]\boldsymbol{w} + b^2 + \mathbb{E}[f_*^2] + 2b\boldsymbol{w}^T\mathbb{E}[\bar{\boldsymbol{g}}] \\ & - 2\boldsymbol{w}^T\mathbb{E}[f_*\bar{\boldsymbol{g}}] - 2b\mathbb{E}[f_*]. \end{aligned}$$

Differentiating $R[\Pi, h]$ with respect to $\boldsymbol{w}$ and b, we get

$$\begin{aligned} \frac{\partial R[\Pi, h]}{\partial \boldsymbol{w}} =& 2\mathbb{E}\left[\bar{\boldsymbol{g}}\bar{\boldsymbol{g}}^T\right]\boldsymbol{w} + 2b\mathbb{E}[\bar{\boldsymbol{g}}] - 2\mathbb{E}[f_*\bar{\boldsymbol{g}}], \\ \frac{\partial R[\Pi, h]}{\partial b} =& 2b + 2\boldsymbol{w}^T\mathbb{E}[\bar{\boldsymbol{g}}] - 2\mathbb{E}[f_*]. \end{aligned}$$

Setting the partial derivatives to $\mathbf{0}$ and solving the equations gives the optimal weight vector and intercept,

$$\widehat{b} = \mathbb{E}[f_*] - \widehat{\boldsymbol{w}}^T\mathbb{E}[\bar{\boldsymbol{g}}], \tag{37}$$

$$\begin{aligned} \widehat{\boldsymbol{w}} =& \left(\mathbb{E}[\bar{\boldsymbol{g}}\bar{\boldsymbol{g}}^T] - \mathbb{E}[\bar{\boldsymbol{g}}]\mathbb{E}[\bar{\boldsymbol{g}}]^T\right)^{-1}\left(\mathbb{E}[f_*\bar{\boldsymbol{g}}] - \mathbb{E}[f_*]\mathbb{E}[\bar{\boldsymbol{g}}]\right) \\ =& \mathrm{Cov}[\bar{\boldsymbol{g}}]^{-1}\mathrm{Cov}[\bar{\boldsymbol{g}}, f_*]. \end{aligned} \tag{38}$$

One can verify that $(\widehat{\boldsymbol{w}}, \widehat{b})$ is indeed minimizing the Bayes risk. Hence,

$$\begin{aligned} \widehat{h}_{\text{S-BLUE}}(\boldsymbol{y}) =& \widehat{\boldsymbol{w}}^T\bar{\boldsymbol{g}} + \widehat{b} \\ =& \mathbb{E}[f_*] + \mathrm{Cov}[\bar{\boldsymbol{g}}, f_*]^T\mathrm{Cov}[\bar{\boldsymbol{g}}]^{-1}(\bar{\boldsymbol{g}} - \mathbb{E}[\bar{\boldsymbol{g}}]). \end{aligned} \tag{39}$$

The terms $\mathbb{E}[f_*], \mathbb{E}[\bar{\boldsymbol{g}}], \mathrm{Cov}[\bar{\boldsymbol{g}}, f_*], \mathrm{Cov}[\bar{\boldsymbol{g}}]$ can all be expressed in closed-form. The closed-form expressions and the details of the computation are given below. Let $\odot$ denote matrix entry-wise

product.

$$\mathbb{E}[f_*] = \mu_*, \tag{40}$$

$$\mathbb{E}[\bar{\boldsymbol{g}}] = \mathbb{E}[\mathbb{E}[\bar{\boldsymbol{g}}|\boldsymbol{\psi}]] = \operatorname{diag}(\mathbb{E}[\boldsymbol{a}])\,\boldsymbol{\mu} + \mathbb{E}[\boldsymbol{b}], \tag{41}$$

$$\begin{aligned}\operatorname{Cov}[\bar{\boldsymbol{g}}, f_*] =& \mathbb{E}[\operatorname{Cov}[\bar{\boldsymbol{g}}, f_*|\boldsymbol{\psi}]] + \operatorname{Cov}[\mathbb{E}[\bar{\boldsymbol{g}}|\boldsymbol{\psi}], \mu_*] \\ =& \operatorname{diag}(\mathbb{E}[\boldsymbol{a}])\,\boldsymbol{k}_*,\end{aligned} \tag{42}$$

$$\begin{aligned}\operatorname{Cov}[\bar{\boldsymbol{g}}] =& \mathbb{E}[\operatorname{Cov}[\bar{\boldsymbol{g}}|\boldsymbol{\psi}]] + \operatorname{Cov}[\mathbb{E}[\bar{\boldsymbol{g}}|\boldsymbol{\psi}]] \\ =& \mathbb{E}[\mathbf{A}\boldsymbol{\mathcal{C}}\mathbf{A} + \varsigma^2\mathbf{M}^{-1}\mathbf{A}^2] + \operatorname{Cov}[\mathbf{A}\boldsymbol{\mu} + \boldsymbol{b}] \\ =& \mathbb{E}[\boldsymbol{a}\boldsymbol{a}^T] \odot \left(\boldsymbol{\mathcal{C}} + \varsigma^2\mathbf{M}^{-1} + \boldsymbol{\mu}\boldsymbol{\mu}^T\right) \\ &+ \operatorname{diag}(\boldsymbol{\mu})\left(\mathbb{E}[\boldsymbol{a}\boldsymbol{b}^T] + \mathbb{E}[\boldsymbol{a}\boldsymbol{b}^T]^T\right) \\ &+ \mathbb{E}[\boldsymbol{b}\boldsymbol{b}^T] - \mathbb{E}[\bar{\boldsymbol{g}}]\mathbb{E}[\bar{\boldsymbol{g}}]^T,\end{aligned} \tag{43}$$

In the terms $\mathbb{E}[\boldsymbol{a}]$, $\mathbb{E}[\boldsymbol{b}]$, $\mathbb{E}[\boldsymbol{a}\boldsymbol{a}^T]$, $\mathbb{E}[\boldsymbol{b}\boldsymbol{b}^T]$, $\mathbb{E}[\boldsymbol{a}\boldsymbol{b}^T]$, the expectations are evaluated entry-wise. For example, entries of $\mathbb{E}[\boldsymbol{a}]$ are given by

$$\mathbb{E}[a_n] = q_0^{(n)} + \sum_{k=1}^{K} q_k^{(n)} \mathbb{E}[a_n | Z_n = k]. \tag{44}$$

Entries of $\mathbb{E}[\boldsymbol{a}\boldsymbol{a}^T]$ are given by

$$\mathbb{E}[a_i a_j] = \begin{cases} \sum_{k=0}^{K} q_k^{(i)} \mathbb{E}\left[{a_i}^2 | Z_i = k\right] & \text{if } i = j, \\ \sum_{k=0}^{K} \sum_{k'=0}^{K} q_k^{(i)} q_{k'}^{(j)} \mathbb{E}[a_i | Z_i = k] & \text{if } i \neq j. \\ \qquad \times \mathbb{E}[a_j | Z_j = k'] & \end{cases} \tag{45}$$

Entries of $\mathbb{E}[\boldsymbol{b}]$, $\mathbb{E}[\boldsymbol{b}\boldsymbol{b}^T]$ and $\mathbb{E}[\boldsymbol{a}\boldsymbol{b}^T]$ can be evaluated similarly. With the above equations, we are able to evaluate $\widehat{h}_{\text{S-BLUE}}$ efficiently. □